

## ❖ Why Windows Server 2012 R2 Parity Storage Spaces Perform Slowly?

The caveat of a parity space is low write performance compared to that of a simple or mirrored storage space, since existing data and parity information must be read and processed before a new write can occur. Parity spaces are an excellent choice for workloads that are almost exclusively read-based, highly sequential, and require resiliency, or workloads that write data in large sequential append blocks (such as bulk backups).

## ❖ What is Journal?

The sequential write performance of parity spaces can be improved by using dedicated journals. A journal is used to stage and coalesce writes to provide resiliency for in-flight I/Os. This journal resides on the same disks as the parity space unless you designate physical disks in the pool as journal disks. The journal is a mirror space and thus resilient by itself. The advantage of dedicated journal disks is a significant improvement in sequential write throughput for parity spaces. Incoming writes are de-staged to the parity space from dedicated disks, thus significantly reducing seeking on the disks used by the parity space. Using two SSDs as a dedicated journal to a three disk parity space, a sequential throughput increase of ~150% was observed under lab conditions. Note that the throughput of the journal disks will now be the overall throughput limit to all parity spaces created on this specific storage pool and you might trade extra capacity for performance. In other words, ensure that dedicated journal disks are very fast and scale the number of journal disks with the number of parity spaces on the pool.

## ❖ What are the minimum requirements?

At least two SSD devices. We recommend three SSD devices, because of the extra SSD can be set as Parallel Rebuild.

## ❖ How to Configure Journaling?

In regards to Windows Server 2012 R2 storage spaces, configuring journaling the SSD drives could be executed with the use of PowerShell. Currently, there is no GUI available that one can use to configure journaling. In order to use SSD drives as journal drives, follow these following steps:

1. Create the storage pool in PowerShell or using the Server Manager Wizard, but do not include the SSDs that you want to be journaled in the pool.
2. After the pool is created, open up PowerShell & execute the following command:

To check all available disk that are able to be pooled

`Get-PhysicalDisk -CanPool $True`

```
PS C:\> Get-PhysicalDisk -CanPool $true
```

FriendlyName	CanPool	OperationalStatus	HealthStatus	Usage	Size
PhysicalDisk12	True	OK	Healthy	Auto-Select	1.46 TB
PhysicalDisk13	True	OK	Healthy	Auto-Select	1.46 TB
PhysicalDisk11	True	OK	Healthy	Auto-Select	1.46 TB

This command creates the variable, which includes all the physical disks that can be added to the storage pool.

`$PDToAdd = Get-PhysicalDisk -CanPool $true`

To enable journaling on the SSDs and add it to the storage pool. **\*Pool1 = name of storage pool.\***

**Add-PhysicalDisk -StoragePoolFriendlyName Pool1 -PhysicalDisks SPDToAdd -Usage Journal**

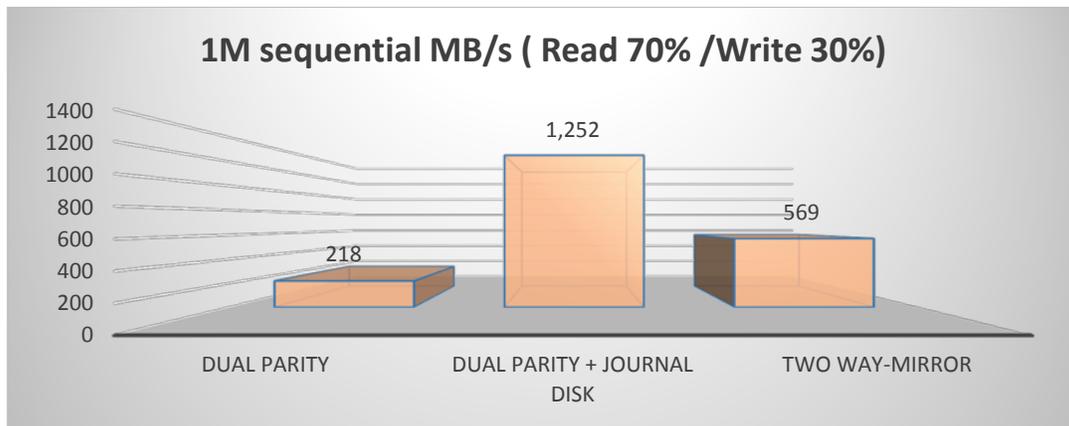
To check the journal configuration was successful

**Get-PhysicalDisk | Sort-Object -Property Usage**

```
PS C:\> Get-PhysicalDisk | Sort-Object -Property Usage
-----
FriendlyName      CanPool      OperationalStatus      HealthStatus      Usage      Size
-----
PhysicalDisk6     False        OK                      Healthy           Auto-Select 1.45 TB
PhysicalDisk0     False        OK                      Healthy           Auto-Select 176.99 GB
PhysicalDisk10    False        OK                      Healthy           Auto-Select 1.45 TB
PhysicalDisk1     False        OK                      Healthy           Auto-Select 1.45 TB
PhysicalDisk4     False        OK                      Healthy           Auto-Select 1.45 TB
PhysicalDisk3     False        OK                      Healthy           Auto-Select 1.45 TB
PhysicalDisk5     False        OK                      Healthy           Auto-Select 1.45 TB
PhysicalDisk2     False        OK                      Healthy           Auto-Select 1.45 TB
PhysicalDisk7     False        OK                      Healthy           Auto-Select 1.45 TB
PhysicalDisk9     False        OK                      Healthy           Auto-Select 1.45 TB
PhysicalDisk8     False        OK                      Healthy           Auto-Select 1.45 TB
PhysicalDisk12    False        OK                      Healthy           Journal     1.45 TB
PhysicalDisk13    False        OK                      Healthy           Journal     1.45 TB
PhysicalDisk11    False        OK                      Healthy           Journal     1.45 TB
```

❖ **How is the performance improve in the sequential write?**

Test Enclosure			
DataON CiB-9470 V12 4U 70 Bay 3.5" – 12G/s SAS			
Disk	HDD: Seagate ST6000NM0034 x70 /FW:E001	HDD: Seagate ST6000NM0034 x70 /FW:E001	HDD: Seagate ST6000NM0034 x70 /FW:E001
	SSD: N/A	SSD: HGST 0B31066 x3 / FW:A290	SSD: N/A
Storage Layout	Dual Parity	Dual Parity + Journal Disk	Two Way-Mirror
Number of Column	17	17	17
Number of Volume	4	4	4
MPIO Policy	Least Blocks	Least Blocks	Least Blocks
Test Tool	IO Meter		
Transfer Request Size	1M sequential MB/s (Read 70% /Write 30%)		
Performance Result	218 MB/s	1,252 MB/s	569 MB/s



Reference sites:

<http://social.technet.microsoft.com/wiki/contents/articles/15200.storage-spaces-designing-for-performance.aspx>

<https://blog.insanegenius.com/2012/09/22/storage-spaces-leaves-me-empty/>