

Offloaded Data Transfer Integration with XCubeSAN

Part Number: QSWP1601A
Published: May 2016
Edition: 1.0



Copyright

© Copyright 2016 QSAN Technology, Inc. All rights reserved. No part of this document may be reproduced or transmitted without written permission from QSAN Technology, Inc.

Edition 1.0 (May 2016)

This edition applies to QSAN XCubeSAN Series. Note that this document was produced based on beta code and some screens may change when it becomes generally available.

QSAN Technology, Inc.

4F., No.103, Ruihu St.,
Neihu Dist., Taipei City 114,
Taiwan (R.O.C.)

Tel: +886-2-7720-2118
Fax: +886-2-7720-0295

Email: sales@qsan.com
Website: www.qsan.com

Notices

This document could include typographical errors or technical inaccuracies. Changes are made to the document periodically. These changes will be incorporated in new editions of the publication. QSAN may make improvements or changes in the products. All features, functionality, and product specifications are subject to change without prior notice or obligation. Document contained herein is subject to change without notice.

The references in this document to non-QSAN websites are provided for convenience only. They do not in any manner serve as an endorsement of those websites. The documents at those websites are not part of the materials for QSAN products. Using those websites is at your own risk.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products.

All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Trademarks

QSAN, the QSAN logo, and qsan.com are trademarks or registered trademarks of QSAN Technology, Inc. All products and trade names used in this document are trademarks or registered trademarks of their respective companies.

Technical Support

Thank you for using QSAN Technology, Inc. products; if you have any questions, please contact QSAN Support. We will reply to you as soon as possible.

- Website: http://www.qsan.com/en/contact_support.php
- Email: support@qsan.com (09:00 GMT+8 ~ 18:00 GMT+8, 09:00 GMT ~ 18:00 GMT)
- Skype ID: qsan.support (09:00 GMT+8 ~ 18:00 GMT+8, 09:00 GMT ~ 18:00 GMT)

Contents

Notices	i
Trademarks	i
Technical Support	i
Offloaded Data Transfer	1
Executive Summary	1
Audience	1
Introduction to ODX	1
Test Environment	2
Test Methodology and Result	4
Performance Comparison	6
Conclusion	7
Apply To	7
Reference	7

Offloaded Data Transfer

Executive Summary

In virtualized and cloud environments, arising escalating data production and demands have created an increasing need for high-speed data transfer. Considering the server and network resource consumption, budgets and limited IT resources, it is necessary to find optimizing existing IT resources inside organizations.

Microsoft® Windows® Offloaded Data Transfer (ODX) enables direct data transfers within compatible storage systems without transferring the data through the host computer. It optimizes system capacity and performance without additional cost or complexity. With ODX, servers are able to offload routine data transfer tasks, reducing the load on servers and storage area networks (SANs).

Offloading the server, ODX utilizes tokenized read/write operations to transfer data at the storage array level. It also greatly improves transfer speeds compared to conventional data transfer methods. QSAN XCubeSAN and AegisSAN series are fully ODX compatible beginning with the SANOS v3.7.0.

Windows Server 2012 ODX and QSAN SAN storage provide a highly efficient and cost-effective solution when deployed together. It also optimizes IT resources and provides agility solution with ever-increasing data.

Audience

This document is applicable for QSAN customers and partners who are familiar with QSAN products. Any settings which are configured with basic operations will not be detailed in this document. If there is any question, please refer to the user manuals of products, or contact QSAN support for further assistance.

Introduction to ODX

Offloaded Data Transfer (ODX) is a feature that has been supported in Windows Server 2012 and later Windows versions for performance improvement purpose with the compatible storage area networks (SANs). Being similar to vStorage APIs for Array Integration (VAAI) in VMware, ODX increases the performance in data copy from one volume to another in the same SAN box. By offloading network flow and CPU loading from on the server, data is would be moved internally in the SAN box, to gain better performance, which is great feature to have in Hyper-V virtualization environment. By moving the data internally in a SAN box instead of transferring via a host, network flows and CPU loadings are offloaded from the servers; this helps to gain better performance and is proved to be a great feature to have in Hyper-V virtualization environment.

In a Hyper-V environment, reducing CPU and network loading means indicates that technician can add more virtual machines or increase the density (more vCPU assigned to mission critical virtual machine) upon hypervisor located on the physical server.

In a traditional file copy/move scenario, when a host connects to with two volumes/LUNs on a storage array, if user tries to copy/move data from one volume/LUN to another, the data transfer follows steps listed below:

- Host reads data from one volume/LUN through network between the host and storage array
- Then the host writes data via the same network to another volume/LUN

ODX speeds up the copy/move operations by offloading the storage array and communicates with the storage using tokens to command reads and writes directly inside the storage array, which ultimately cut down CPU cycles on the host.

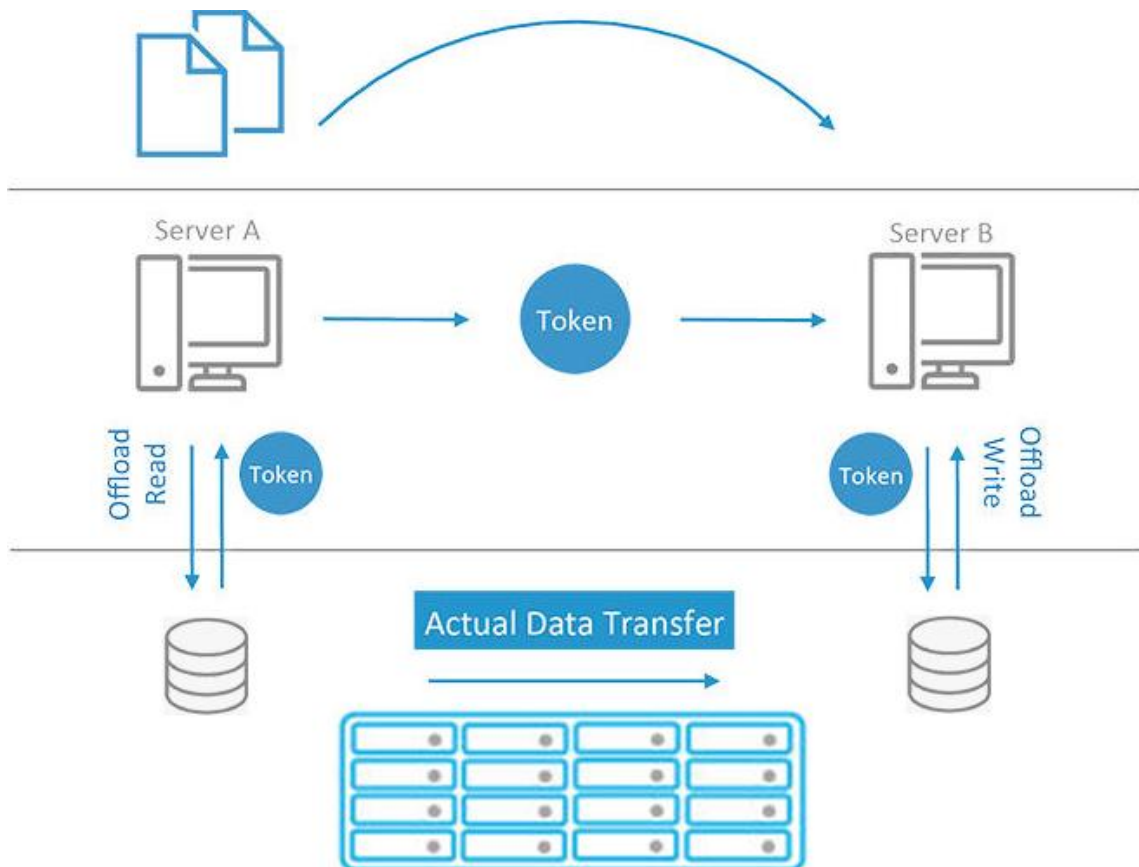


Figure 1 ODX Architecture

Test Environment

Host

- Operating System: Windows Server 2012 R2 Datacenter Edition

Storage

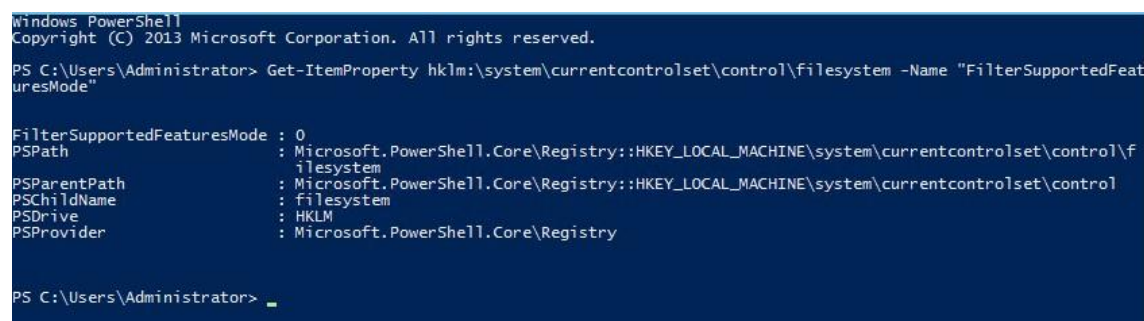
- Model: QSAN XCubeSAN XS5216-S
- Firmware version: V4.0.0
- Volume/LUN: 100GB volume x 2 (named as VD-a and VD-b)

Test file

- 12GB video file compressed with WinRAR

Before starting the test, please make sure that the ODX function is enabled on the host; please check the value of ODX through the following commands on PowerShell:

```
C:\> Get-ItemProperty hklm:\system\currentcontrolset\control\filesystem -Name
"FilterSupportedFeaturesMode"
```



```
Windows PowerShell
Copyright (C) 2013 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> Get-ItemProperty hklm:\system\currentcontrolset\control\filesystem -Name "FilterSupportedFeaturesMode"

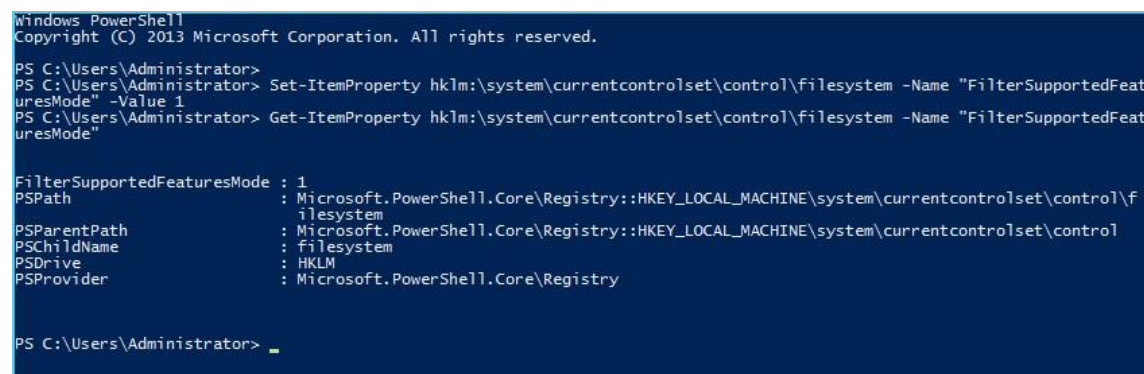
FilterSupportedFeaturesMode : 0
PSPath                      : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\system\currentcontrolset\control\filesystem
PSParentPath                : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\system\currentcontrolset\control
PSChildName                 : filesystem
PSDrive                     : HKLM
PSProvider                  : Microsoft.PowerShell.Core\Registry

PS C:\Users\Administrator> _
```

Figure 2 ODX Status

Command to Disable ODX

```
C:\> Set-ItemProperty hklm:\system\currentcontrolset\control\filesystem -Name
"FilterSupportedFeaturesMode" -Value 0
```



```
Windows PowerShell
Copyright (C) 2013 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> Set-ItemProperty hklm:\system\currentcontrolset\control\filesystem -Name "FilterSupportedFeaturesMode" -Value 1
PS C:\Users\Administrator> Get-ItemProperty hklm:\system\currentcontrolset\control\filesystem -Name "FilterSupportedFeaturesMode"

FilterSupportedFeaturesMode : 1
PSPath                      : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\system\currentcontrolset\control\filesystem
PSParentPath                : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\system\currentcontrolset\control
PSChildName                 : filesystem
PSDrive                     : HKLM
PSProvider                  : Microsoft.PowerShell.Core\Registry

PS C:\Users\Administrator> _
```

Figure 3 Disable ODX

Command to Enable ODX

```
C:\> Set-ItemProperty hklm:\system\currentcontrolset\control\filesystem -Name
"FilterSupportedFeaturesMode" -Value 1
```

```
Windows PowerShell
Copyright (C) 2013 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> Set-ItemProperty hklm:\system\currentcontrolset\control\filesystem -Name "FilterSupportedFeaturesMode" -Value 0
PS C:\Users\Administrator> Get-ItemProperty hklm:\system\currentcontrolset\control\filesystem -Name "FilterSupportedFeaturesMode"

FilterSupportedFeaturesMode : 0
PSPath                      : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\system\currentcontrolset\control\filesystem
PSParentPath                : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\system\currentcontrolset\control
PSChildName                 : filesystem
PSDrive                     : HKLM
PSProvider                  : Microsoft.PowerShell.Core\Registry

PS C:\Users\Administrator> _
```

Figure 4 Enable ODX



NOTE:

QSAN series products support ODX by default; user doesn't need to do anything to enable it.

Test Methodology and Result

Without ODX

Copy the file (12GB) from Volume A to Volume B on the host through a single GbE NIC

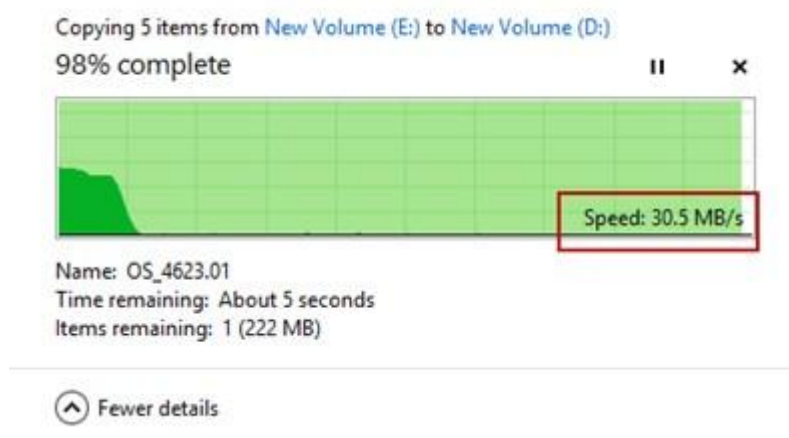


Figure 5 Test Result without ODX

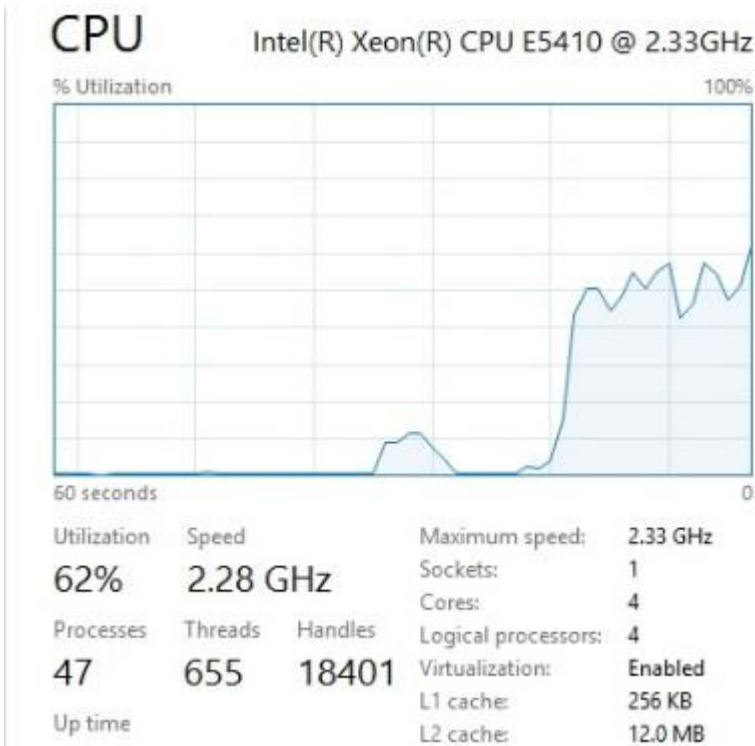


Figure 6 CPU Utilization without ODX

With ODX

Copy the file (12GB) from Volume A to Volume B on the host through a single GbE NIC

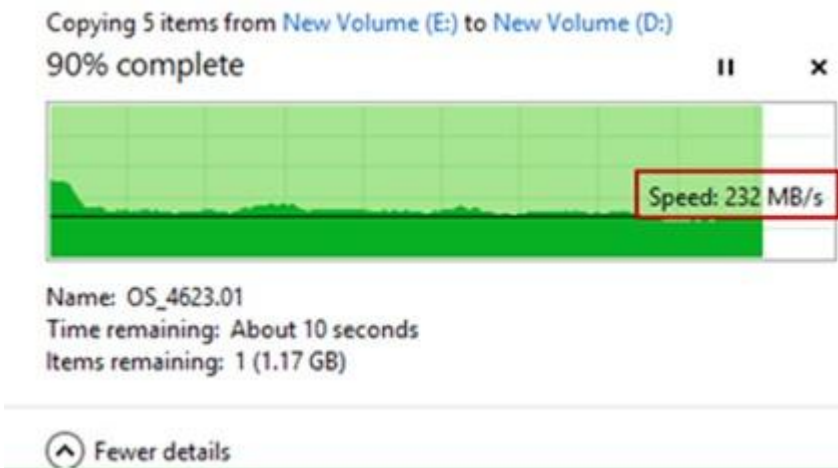


Figure 7 Test Result with ODX

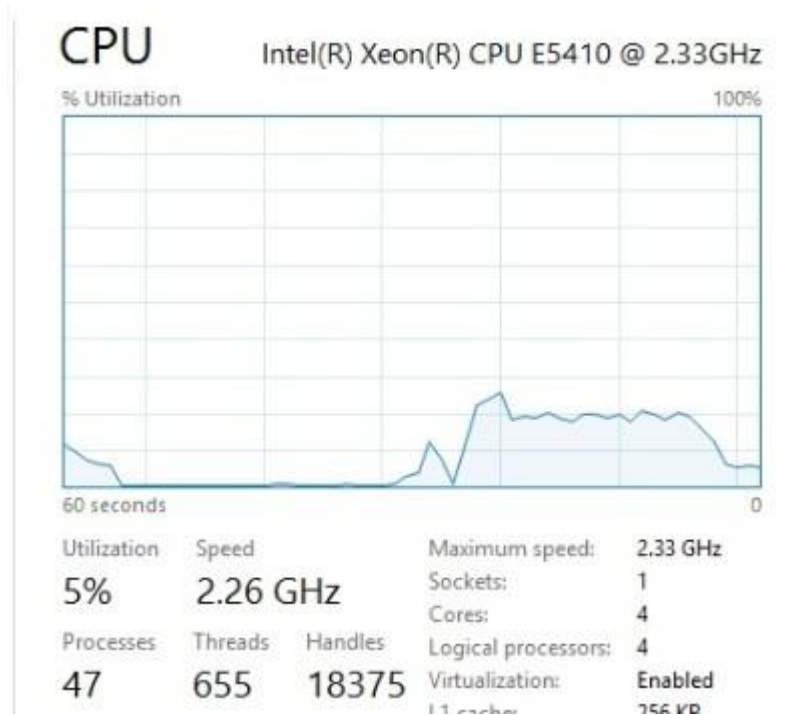


Figure 8 CPU Utilization with ODX

Performance Comparison

Time Consuming and Throughput

Without ODX enabled, it costs 490 seconds and 30.5 Mb/s throughput only. With ODX enabled, the time consuming reduces to 56 seconds and throughput increases to 232 Mb/s. In total, it has increased around 8 times.

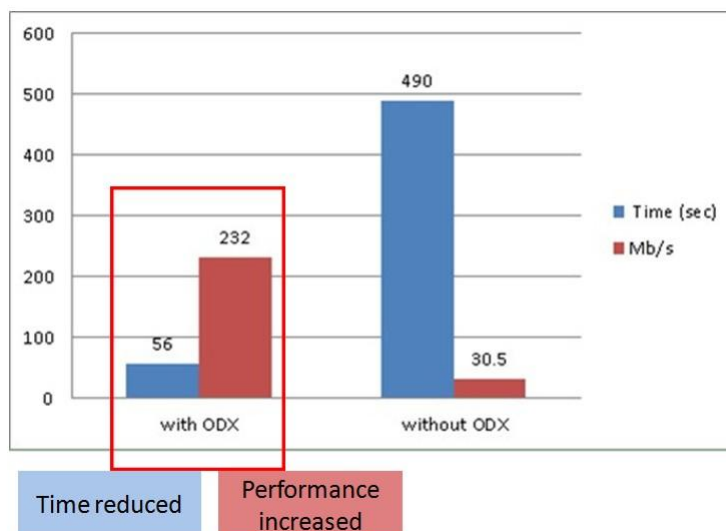


Figure 9 Performance Comparison

Conclusion

ODX is helpful of improving performance when trying to move/copy data from one volume to another which is created in the same storage array and is connected to the same server (or another server which is in the same cluster group as the source one). Qsan storage array supports ODX feature, for detailed information, please refer to Apply To section below pertaining to the exact models and firmware versions.

Apply To

- XCubeSAN FW 1.0.0 (SANOS 4.0)
- AegisSAN Q500 FW 3.7.0
- AegisSAN LX FW 3.7.0
- AegisSAN V100 FW 3.7.0

Reference

Microsoft Documentations

- [Microsoft Developer Resources – Offloaded Data Transfer](#)
- [Microsoft TechNet – Windows Offloaded Data Transfers Overview](#)
- [Microsoft TechNet – Deploy Windows Offloaded Data Transfers](#)