White Paper

VMware vSphere Storage API

Integration with XCubeSAN

Part Number: QSWP1602A
Published: May 2016
Edition: 1.0

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

VMware VAAI

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.

VMware® vSphere® Storage APIs for Array Integration (VAAI) 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 VAAI, servers are able to offload routine data transfer tasks, reducing the load on servers and storage area networks (SANs).

Offloading the server, VAAI utilizes 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 VAAI compatible beginning with the SANOS v3.7.2.

VMware ESXi 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 VMware VAAI

VMware vSphere Storage APIs for Array Integration (VAAI) is an application program interface (API) framework from VMware that enables many storage tasks, e.g. Thin Provisioning, Full Copy, Block Zero and Hardware Assisted Locking. VAAI is supported and fully integrated in QSAN XCubeSAN series products with VMware ESXi version 5.x or later. Resources on VMware ESXi server(s) are saved by this integration when performing storage-related tasks between the hypervisor of VMware ESXi and QSAN XCubeSAN series products.

VAAI was introduced in VMware vSphere 4.1 with the following features implemented for achieving offload capabilities:

- Full Copy or Hardware Assisted Move
- Block Zero or Hardware-Assisted Zero

- Hardware Assisted Locking or Atomic Test and Set (ATS)

And Thin Provisioning was introduced in VMware vSphere 5.x. Detailed explanations of these features are presented as following:

Thin Provisioning

VMware vSphere 5.x implements some VAAI enhancements for the scenario which uses storage-based Thin Provisioning feature, which is also supported in QSAN XCubeSAN series products. Two main enhancements of VAAI Thin Provisioning are:

- Dead Space Reclamation (also known as UNMAP)
- Out of space conditions

Dead Space Reclamation

Traditionally, when a storage volume/LUN was mounted as a datastore, and there were virtual machines stored in the datastore, if any of virtual machines were deleted or migrated, the storage space [(m)-3(acq1 [am i1-.n5)5(ead)-5(Sp)r

as though the process of writing zeros has been completed. QSAN XCubeSAN finishes the zeroing out internally. Please refer to Figure-2 below, which shows the operation and process how Block Zero is performed between VMware ESXi server andQSAN XCubeSAN.

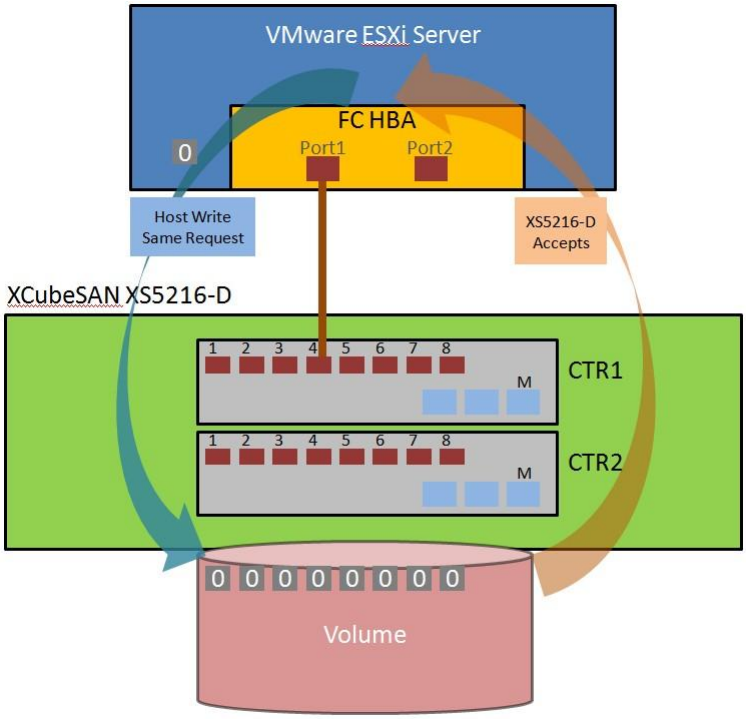

Figure 2 VAAI Block Zero

Hardware Assisted Locking

Hardware Assisted Locking, also called ATS (Atomic Test and Set), provides an alternate method to protect the metadata for VMFS cluster file systems and improve the scalability of large ESXi servers which are sharing a VMFS datastore. ATS helps locking blocks in a volume/LUN instead of the whole volume/LUN which is added as a datastore in a VMware ESXi server.

Effective Operations

- Create a VMFS datastore
- Expand a VMFS datastore onto additional extents
- Power on a virtual machine
- Acquire a lock on a file
- Create or delete a file
- Create a template
- Deploy a virtual machine from a template
- Create a new virtual machine
- Migrate a virtual machine with vMotion

- Grow a file (e.g., a snapshot file or a thin-provisioned virtual disk)

Advantage

Hardware Assisted Locking (or ATS) gives a much more efficient way to avoid retries for getting a lock when multiple VMware ESXi servers are sharing the same datastore. The lock mechanism is offloaded to the storage array, and the storage array executes the lock at a granular level. This is helpful in scalability when a datastore is shared in a VMware cluster environment without compromising the integrity of metadata in the VMFS shared storage pool.

Theory

Previously, VMware had a similar mechanism of locking a virtual machine to prevent from being run on, modified by more than one VMware ESXi server in the same time. This mechanism was built on the use of SCSI RESERVE and RELEASE commands. This protocol calls the unique access to an entire volume/LUN for the reserving ESXi server until this ESXi server sends a release. Under the protection of SCSI RESERVE command, an ESXi server can update metadata records on storage array to reflect the usage of storage array without being interfered by any other ESXi servers which also call the same portion of the same storage array. Please refer to the Figure-3 shown below which shows the entire structure in this scenario, and which impacts the overall performance in the whole cluster VMware ESXi environment. Situations of slow performance caused by huge quantity of RESERVE and RELEASE commands are unacceptable in VMware cluster environment which accesses the shared datastore from different virtual machines exponentially every day.

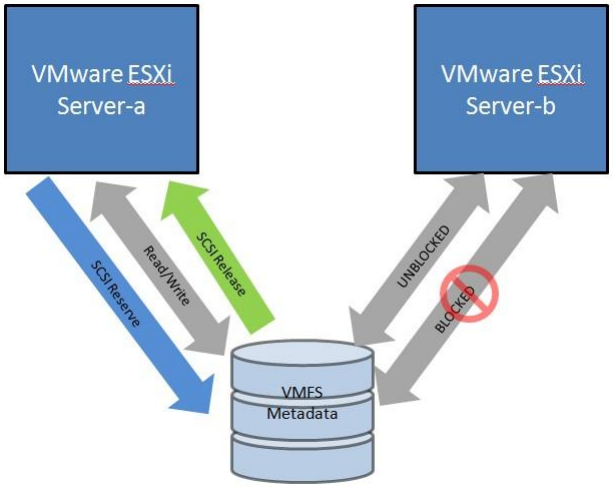

Figure 3 VAAI Hardware Assisted Locking

With VAAI, Hardware Assisted Locking delivers a more granular method of protecting the VMFS metadata than SCSI RESERVE and RELEASE commands. Hardware Assisted Locking uses a storage array ATS capability to enable a fine-grained block-level locking mechanism as shown below in Figure-4. First of all, Hardware Assisted Locking replaces the sequence of RESERVE, READ, WRITE and RELEASE SCSI commands with a single SCSI COMPARE AND WRITE (CAW) request for an atomic read-modify-write operation, based on the presumed availability of the target lock. Then, this new request only requires exclusion of other accesses to the target locked

block, not the entire VMFS (which is volume/LUN) which contains the requested lock. This locking metadata update operation is used by VMware when the state of a virtual machine changes. This may be a result of the virtual machine being powered ON or OFF, or modifying the configuration of a virtual machine, or even migrating a virtual machine from one ESXi server to another through vMotion.

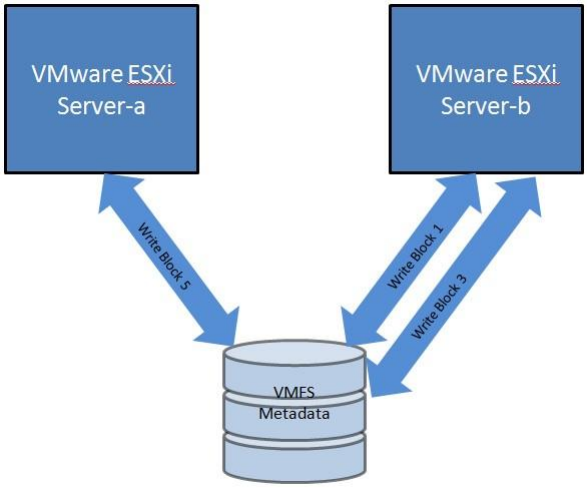

Figure 4 VAAI Hardware Assisted Locking

Hardware Acceleration Support Status

The status of Hardware acceleration can be observed after adding any storage volume/LUN through VMware vSphere Client. Please navigate to Configuration \ Hardware \ Storage and click Datastores View, check the Hardware Acceleration column shown behind each added datastore, as Figure-5 shown below.

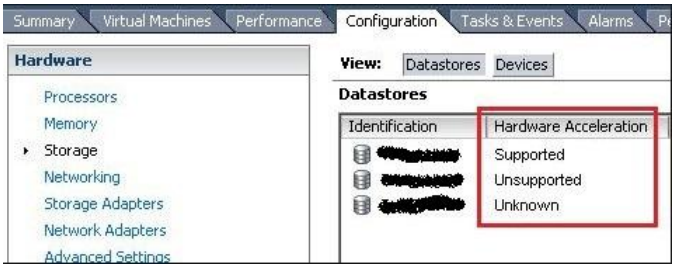

Figure 5 VAAI Hardware Acceleration Support Status

Table 1 Hardware Acceleration Status values

Status Value	Description
Supported	Storage devices support VAAI
Unsupported	Storage devices do not support VAAI
Unknown	Local datastores

Test Environment

Here we use an example which we set up an environment that connects a VMware ESXi server with QSAN XCubeSAN XS5216-D storage array for demonstrating VAAI functionalities.

Architecture

Please refer to the Figure-6 shown below regarding to the FC connection between XCubSAN XS5216-D storage array and VMware ESXi server. A brief and simple environment is made in this example. The real architecture in customers' environment could be more complex.

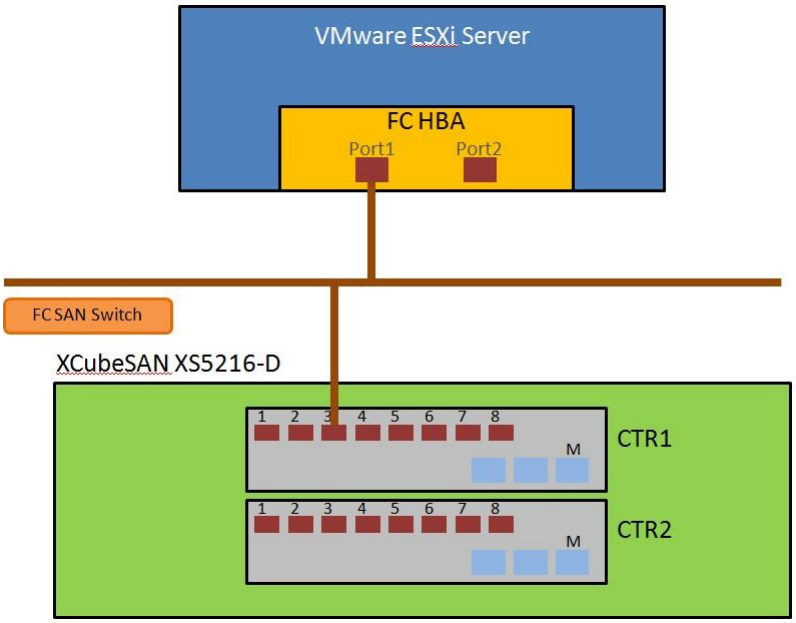

Figure 6 VAAI Architecture

Storage Configuration

This test was made by adding two FC volumes/LUNs from XS5216-D storage array as two VMFS datastores separately within one VMware ESXi server, in order to simulating VM clone and storage vMotion function. Figure -7 gives the idea of how the pools and volumes were created. The Cache Mode of these volumes was set as WT (Write Through, the cache of storage array is set as OFF on this volume) when verifying the time consumed for the best data protection. The Cache Mode can be modified online without stopping services, please navigate to STORAGE MANAGEMENT\Volumes page on web UI of XCubeSAN XS5216-D storage array, click the arrow-down button in front of the created volume, select Change Volume Properties for adjusting. Please remember to change the Cache Mode back to WB (Write Back, the cache of storage array is set as ON on this volume) after the verification is finished.

The same architecture and storage configuration can also be made through iSCSI connections over network.

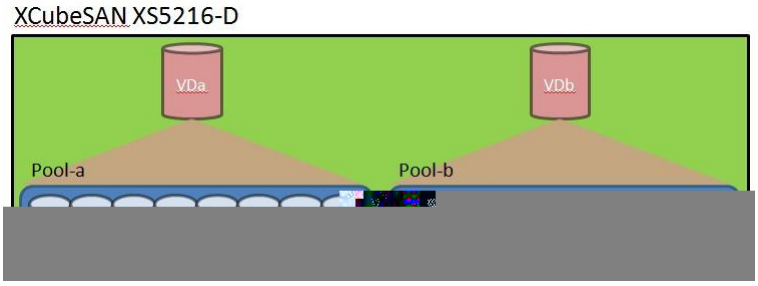

Figure 7 VAAI Storage Configuration

Test Methodology and Result

This section describes test methodology and test result.

Full Copy

Steps to Verification

1. Created a virtual machine with a 200GB Thick Provision Lazy Zero virtual disk on a VMFS datastore which was made by a FC volume/LUN from XS5216-D storage array. The actual storage consumption on the datastore was around 77GB.
2. Migrated or cloned the virtual machine from this datastore to another datastore (which was made by another FC volume/LUN from the same XS5216-D storage array.
3. Observed the time cost to migrate or clone the virtual machine.
4. Repeated step from 1 to 3 above, with VAAI OFF, and compared the time cost.
5. Table-2 shown below provides the results between VAAI ON and VAAI OFF after the tests were finished.

Table 2 Time Cost for Full Copy

Full Copy use case	VAAI Off	VAAI On
Storage vMotion	26 minutes 56 seconds	6 minutes 05 seconds
Virtual Machine Clone	25 minutes 50 seconds	5 minutes 59 seconds

Completion

It provides 77.4% faster when testing storage vMotion in comparison with the scenario of VAAI OFF when VAAI is ON; and, improved around 76.8% faster performance when testing virtual machine clone.

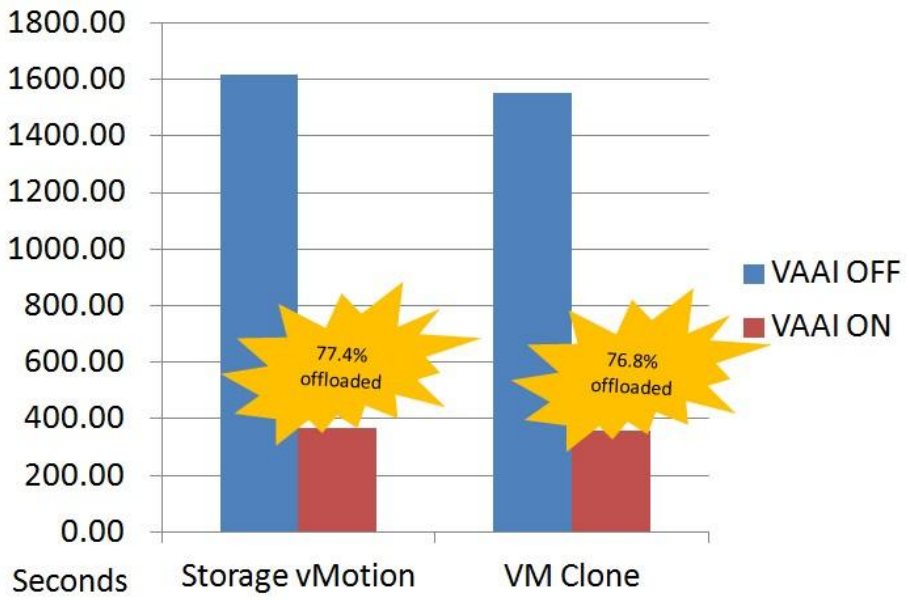

Figure 8 VAAI Test Result with vMotion

Block Zero

Steps to Verification

1. Measured the time taken to create a 200GB Thick Provision Eager Zero virtual disk on a virtual machine.
2. Repeated the same step above in comparison with VAAI OFF.
3. Table-3 shown below provides the results between VAAI ON and VAAI OFF after the tests were finished.

Table 3 Time Cost for Block Zero

Block Zero use case	VAAI Off	VAAI On
Thick pool volume	9 minutes 6 seconds	4 minutes 0 seconds

Completion

The performance got 56% faster when VAAI was enabled and tried to create a 200GB Thick Provision Eager Zero virtual disk which stores in a Thick pool volume on XCubeSAN XS5216-D storage array.

Thin Provisioning

Steps to Verification

1. Created a Thin Provisioning volume/LUN on XS5216-D storage array, 200GB was allocated.
2. Created a VMFS datastore on the connected VMware ESXi server.

3. Created a virtual machine based on this VMFS datastore, type of Disk Provision was set as Thin Provision, size was set as 100GB.
4. Generated some data on the virtual machine, around 50GB, observed the capacity consumed on VMFS datastore, 50GB of 200GB was used.
5. Performed storage vMotion with VAAI ON to migrate the virtual machine to another VMFS datastore.
6. Observed the capacity consumed on the source VMFS datastore again, you shall discover that the used capacity is around 0GB of 200GB.
7. However, checked the Available Capacity (GB) on web UI of XS5216D after the migration of virtual machine, it may still show 50GB was occupied, that is because the granularity in QSAN XCube XS5216D storage array is 1GB, only a continuous 1GB as zero blocks can be reclaimed.
8. Please create a new virtual machine with Thick Provision Eager Zero on this VMFS datastore, and delete it after the creation, then execute Space Reclamation on XS5216D storage array, the space shall be able to be reclaimed. You may find the Space Reclamation function on Storage Management\Volumes page on web UI, click on the arrow-down button in front of the created volume then select Space Reclamation.

Completion

The granularity supported in Thin Provisioned pool in QSAN XCubeSAN series products is 1GB, though space reclamation can be enabled when a volume is created, sometimes it still needs to manually fill zero blocks from server so that the unused blocks can be filled as zero and reclaimed.

Conclusion

The integration of VAAI in QSAN XCubeSAN series products provides lots of benefits of increasing performance as well as the management of storage array. The main features are:

- The Full Copy feature accelerates the storage vMotion or virtual machine clone operations and vastly reduces the usage of resources while performing these operations by offloading the operation from VMware ESXi server to storage array itself.
- The Block Zero feature speeds up the deployment of Thick Provision Eager Zero virtual disks by offloading the duplicated and repetitive zero of large numbers of blocks to the QSAN XCubeSAN platform, helps to free the resources of VMware ESXi server for other tasks.
- The Hardware Assisted Locking feature delivers much more efficient methods to prevent the retries of getting a lock when multiple eThe delinuand