TANEJA GROUP
TECHNOLOGY ANALYSTS

TECHNOLOGY BRIEF

THE PROMISE OF VM-CENTRIC STORAGE AND VVOLS

Tintri VMstore Delivers the Future Promise Now

JUNE 2015

THE NET NET

The din surrounding VMware vSphere Virtual Volumes (VVols) is deafening. It started in 2011 when VMware announced the concept of VVols and the storage industry reacted with enthusiasm and culminated with its introduction as part of vSphere 6 release in April 2015. Viewed simply, VVols is an API that enables storage arrays that support the functionality to provision and manage storage at the granularity of a VM, rather than LUNs or Volumes or mount points, as they do today. Without question, VVols is an incredibly powerful concept and will fundamentally change the interaction between storage and VMs in a way not seen since the concept of server virtualization first came to market. No surprise then that each and every storage vendor in the market is feverishly trying to build in VVols support and competing on the superiority of their implementation.

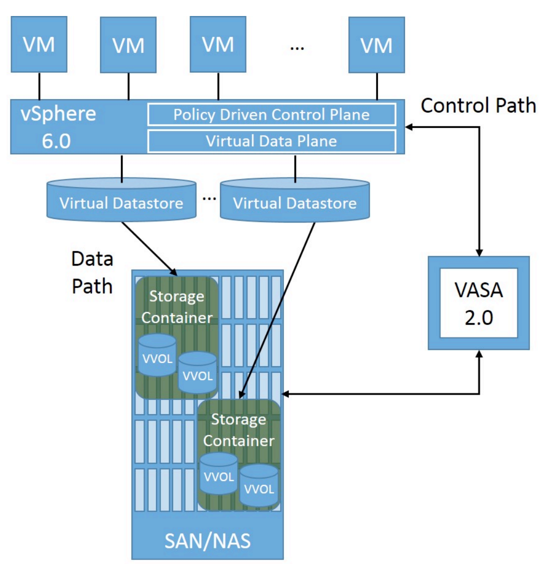
Yet one storage player, Tintri, has been delivering products with VM-centric features for four years without the benefit of VVols. How can this be so? How could Tintri do this? And what does it mean for them now that VVols are here? To do justice to this question we will briefly look at what VVols are and how they work and then dive into how Tintri has delivered the benefits of VVols for several years. We will also look at what the buyer of Tintri gets today and how Tintri plans to integrate VVols. Read on…

PRIMER ON VMWARE VIRTUAL VOLUMES (VVOLS)

The Basic Rationale

VMware VVols are designed to overcome a fundamental deficiency in the way shared storage works with virtual machines today. The issue has to do with the fact that VMware works at the granularity of a VM and storage at the granularity of a LUN/Volume or a Mount Point (for brevity's sake we will refer to all of these as a LUN). This disparity wreaks havoc in many areas: Given that most storage arrays today have a restricted number of LUNs capability, a single LUN often includes 10's of VMs, making measurement of performance or other factors impossible at the VM level. For example, if a rogue VM starts usurping a large percentage of the resources, it is hard to identify it because the only management visibility available is at the LUN level. In addition, since data services such as replication or cloning or snapshots, are implemented at the LUN level, one cannot perform these services at a VM level. This results in a

VVols live within an ecosystem provided by vSphere 6.0, VASA 2.0 and each separate vendor's specific implementation.



87 Elm Street, Suite 900 · Hopkinton, MA 01748 · T: 508.435.2556 · F: 508.435.2557
www.tanejagroup.com

waste of compute, storage and network resources when only a single VM needs a particular data service.

VVols aims to address this mismatch by aligning the unit of consumption with the unit of management. This means resources are allocated at a VM level; data services are performed at a VM level; monitoring is done at a VM level and data is analyzed at a VM level. This level of visibility transforms the VMware environment from opacity to transparency and results in significantly shorter times to manage and a noticeable reduction in costs. Note that these are the end goals for VVols even though the current implementations are not there yet.

VVol Architecture

To bridge from the shared storage array world into the VM world, there are several constructs and layers whose job it is to logically partition the task. Within the array, the storage admin creates Storage Containers (SC). Storage Containers are a logical concept that lives above the LUN level and where each VVol is stored. A VVol is needed for each of the files within a VMware environment: a VM's config, swap, data, any application specific files and, when executed, snapshots. Multiple SCs are set up where each provides a different mix of performance and data service capabilities (flash, cloning, snapshots, replication…). SC's are presented to the vSphere environment as a one-to-one mapping to a Virtual Datastore. VVols are a new container within Virtual Datastores, and need to be placed correctly so the VM has access to the desired capabilities.

The VM admin has the responsibility of understanding the required capabilities and ensuring each VVol is placed into the right Virtual Datastore/SC. In order to perform this task the admin needs to know what SCs exist, and the capabilities of each. The vSphere API for Storage Awareness (VASA – the 2.0 version is required to use the VVol capability) is the control path where the array provider ensures each SC and its attributes are communicated to the vSphere environment. So while the basic information is available to the Virtual Control Plane and can be referenced, this is only half of a practical solution; having to do the assignment manually would be very time consuming and error prone. A Policy Based Control Plane within vSphere 6.0 enables automated assignment and management. The VM admin uses the storage policy-based management (SPBM) interface and assigns a storage policy to each VM by choosing the desired capabilities from a simple list that is dynamically populated through VASA. The right VVols are then created and placed into the correct SC by the storage array.

If needs of a VM change, the SPBM can be used to specify another capability or performance level, and if the changes can take place without having to create a new environment and do a vMotion, as was the case in the past, SPBM will leave the VM in place an apply a new policy. If the requested capability only exists in a different SC than the VVols associated with the VM are currently in, the array is responsible for moving them appropriately. All of the performance related information is specified at the VM level. The VVols abstraction is a software-based capability that enables changes in a more flexible manner than having to be aware of and manage the specifics of the physical array.

VVols Implementation

VVols is not a new protocol but rather it is an API that is part of VASA 2.0. The same storage protocols (FC, iSCSI, NFS, FCoE) are supported in vSphere 6.0 as were available previously. A new construct, a Protocol Endpoint (PE) is provided by each array vendor. PEs are a transport mechanism that connects VMs to their VVols on demand through the chosen storage protocol. For block storage this can potentially relieve LUN configuration limitations as a PE can support a very large number of VVols. For file storage the PE is a discoverable mount point and each VVol lives within its own file object.

VVols are designed to eliminate the mismatch between the VM administrator's view (VM-centric) and the storage administrator's view (LUN-centric). It bifurcates the responsibilities differently in such a way that the VM administrator is now able to provision and manage storage at a VM-level. All is not nirvana, however, as VVols is a 1.0 API and each vendor is choosing how and when to support its present and future capabilities in a manner that fits with their architecture and capabilities. Even if the VVols capabilities were complete (which they are not) the storage vendor still has to decide which VVol capabilities can be implemented without an architectural change of the storage array. Even beyond the limitations built into a given array the end user still has to expend significant effort in order to get VVols working for them. There are many misunderstandings associated with VVols, some of which are listed below.

Misunderstandings Surrounding VVols

One common misunderstanding is that all one needs to get VVols support is to upgrade to vSphere 6. In reality, one needs to upgrade the storage array and the vSphere release. As mentioned above enabling VVols in a storage array is a non-trivial exercise in and of itself and has to be done by the storage array vendor. With the updated storage array and a vSphere 6 release the end user then needs to follow the procedure outlined below. The typical process is to schedule downtime, complete the installs and upgrades of all the required SW and HW, and then to verify that all is again working as it was before the upgrade started. The environment is then configured, policies defined and new VVol-based VMs instantiated. Existing VMs that need to be VVol-enabled can then be migrated into the new environment, usually, one at a time. As with any major environment upgrade, the process can be complicated and time consuming and should be investigated and planned accordingly.

A major consideration in any environment is how it scales, and VVols do not intrinsically address scale. How an array scales depends on many factors and one of those factors is the number of VVols it can support. To determine how many VMs an array with a VVol limit of 10,000 can support, one must understand how many VVols are needed per VM. Typically one VM is made up of a config file, a swap file and at least one VMDK file (more complicated VMs can have more than one VMDK file). In the world of VVols, each of these files is represented by a VVol, suggesting a minimum of three VVols per VM. If this were the end of the story this array would support a total of 3,333 VMs, which is not an insignificant number. However, in reality the number of supported VMs is much lower, depending on the number of snapshots that need to be stored. This is because each snapshot creates a minimum of one VVol. If snapshots were taken every 15 minutes and a week's worth kept on the array, one would add 672 VVols, for a total of 675 VVols per VM. This instantly reduces the maximum number of VMs supported by this array to about 15, an unacceptably low number for most IT environments. The general impression in the market is that VVols suddenly improve the scalability of an array. In reality, even an array with a 10,000 VVol limit can be very confining.

Another misunderstanding is that VVols provide per-VM QoS and performance isolation. Unquestionably VVols are a major step forward, but do not in and of themselves provide VM-level QoS or isolation. That must be implemented by the underlying array. The level of QoS that the array provides is not changed by the existence of VVols.

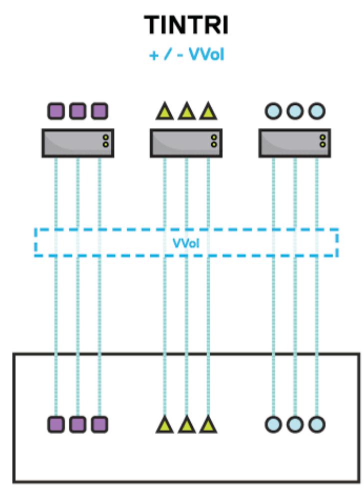
For the most part vSphere 6 and VVols do not support server-side replication and QoS. VVol-enabled arrays can surface up array-side replication and QoS, if they support those functions. But since VASA 2.0 does not support VVol replication or QoS in the spec, vSphere can take no action on this capability and simply returns control to the array. Similarly, VVols are not interoperable with SRM (VMware KB article 2112039). This means SRM cannot use VMs stored using VVols. There are many other such limitations at the moment with vSphere 6, VVols and various applications that run with vSphere.

Perhaps the greatest misunderstanding is that VVols, in effect, level the playing field between competitive storage arrays: that simply by supporting VVols, an inadequate array suddenly becomes equal to another that had a superior feature set. This cannot be further from the truth. Several innovative vendors had delivered on the promise of VM-centricity well before VVols came to market. Some vendors also had features such as QoS implemented at VM level of granularity. Such vendors will continue to enjoy the differentiation, even as other vendors rush to support VVols. So while it is true that VVols provide a VMware-sanctioned method of enabling VM-centricity it is entirely wrong to think that all arrays will therefore have equivalent features. Vendors that have invested in VM-centric storage will retain their advantage.

TINTRI VMSTORE

Shipping for over 4 years, Tintri's VMstore was built from the ground-up based on the industry's first VM-aware storage architecture. The array uses existing support in vSphere for NFS as the basic connection mechanism, but this is where the similarity with other LUN and volume-based arrays stops. The entire array is presented as a single virtual datastore and supports different classes of service for each VM. When contrasted against LUN and volume-based arrays there is no comparison. However, even when contrasted against many VVol-supported arrays this straightforward approach wins out in that storage administrator does not have to create different storage containers, each with a specific combination of performance and feature set. So how does VMstore avoid this complication and still support different service levels? It is due to the architectural focus on the VM.

VMstore maps directly to the VM and vDisk.



VMstore can analyze and track each IO request to individual VMs and vDisks. The protocol manager at the top of VMstore effectively splits the VMs and vDisks into separate IO "lanes" that allows critical system resources in the Tintri file system to be allocated to individual, rather than groups of VMs. VM-level isolation and active management is simply how VMstore works. The end result is that consistent performance is the norm without manual tuning or administrative oversight. When this is not sufficient due to high scalability installations or "overly demanding" VMs, VM-level QoS provides the ability to specify minimum and/or limit maximum resource consumption. As a result "noisy neighbors" (bursty, demanding database applications or rouge VMs) cannot steal resources from latency sensitive remote desktops in VDI deployments, for instance.

Each component of latency is displayed at a VM level permitting the right issue to be identified and addressed.

In order to provide the highest performance, VMstore uses flash as the primary storage "tier"; on its hybrid arrays it uses high capacity HDD as well in order to provide the best cost effectiveness. The trick is to ensure that SSDs are used as much as possible while HDDs are used only to store "cold" data. Tintri's working set analysis uses sophisticated IO profiling that separates the active data from its dormant brethren. This ensures that the high performance (and higher cost) SSDs are used to maximum advantage delivering near "all-flash" performance.

After making sure that only the data that needs to be in flash is in flash, cost effectiveness is further enhanced by making sure that no capacity is wasted. This is done in two ways: only store what is needed, and make sure what is stored is as small as possible. All data must be stored, but it doesn't need to be stored twice just because it happens to exist in more than one VM. It is well known fact that much of the data that needs to be stored is duplicated between more than one VM; sometimes duplicated in practically *every* VM. VMstore uses in-line deduplication to detect when that is the case and stores multiple references to a single instance of the data rather than multiple copies. In-line compression also makes sure that all the data is as compact as it can be.

With a focus on both performance and capacity efficiency, Tintri has produced a system that boasts 100TB effective storage, delivering 140K IOPs and supporting 3500 VMs in just 4U in the VMstore T880 system.

Perhaps the most advantageous benefit of VMstore, beyond delivering what VVols based solutions promise, is the extreme ease of management and visibility it provides. An example of the visibility is shown in the screen shot to the left that shows each component of latency. In an instant the administrator knows how much latency is being introduced by the host, network, storage (which in turn is broken down into "contention", flash and disk) and finally, "throttle." This last component shows up if the VM-level QoS is enforcing a maximum limit and this limit is being reached. Detailed per-VM performance visibility like this allows the admin to easily locate and resolve issues with a level of precision that in unprecedented in the storage industry.

VMstore is endowed with smart-store capabilities that intelligently manages individual VMs. In December of 2013, Taneja Group completed a hands-on examination to determine the benefits of a virtualization-specific storage solution. Beyond validating the benefits described above, we estimated a 60X estimated annual management time and effort reduction from Tintri storage as compared to traditional offerings. In essence, with an architecture that guarantees VM performance, Tintri reduces storage interaction to the point that it almost disappears, and every storage operation suddenly just becomes a VM operation – you no longer manage a data store, you manage VMs. At the time, Tintri called their solution "zero-management" and while not truly zero touch it was and still is the simplest management we had ever tested. That report is available from Tintri's or Taneja Group's website.

A final consideration is that with Tintri VMstore you can take advantage of a VM-centric solution without having to upgrade to vSphere 6.0, as it supports vSphere 4.x through 6.0 versions including integration into vCenter. In addition, while VVols only work with vSphere, Tintri supports all other major hypervisors including Microsoft HyperV, Citrix XenServer, Red Hat Enterprise Virtualization and OpenStack environments. Therefore, whether you are an ESXi, HyperV, Xen or KVM shop you are covered.

It is clear from the above discussion that Tintri VMstore has solved all the fundamental issues that VVols are designed to solve, at least today and in the near future, in its own way. Tintri plans to introduce VVol support later this year. But given the current functionality in VVols 1.0, VASA 2.0 and vSphere 6, it is clear that Tintri will continue to enjoy vastly superior functionality in their native version for the foreseeable future. But the buyer of Tintri should also know that over time Tintri will make a non-disruptive migration to VVols available.

TANEJA GROUP OPINION

VVols are single-mindedly the best thing that has happened to storage since the dawn of VMware. The mismatch between storage working at the granularity of LUNs/Volumes/Mount Points and VMware working at the granularity of VMs has caused severe pain for both the storage administrator and the VMware administrator. With VVols 1.0, this mismatch starts to disappear and provisioning and management of storage starts to become easier.

Given this, VVols are here to stay. However, the reality is also that we have just entered the era of VVols 1.0. Not all functionality that vSphere 6 is capable of is built into VVols. Many VMware products do not yet work with VVols. At the same time, the storage industry is just bringing out their first rev of VVols supported arrays. Many have plans to support some but not all features of VVols. The industry is exactly where it is, each time it does a 1.0 release. In our view, it will take 18 to 24 months to get additional functionality of VVols included in some of the popular arrays in the market.

One can wait for that panacea or one can seriously look at Tintri, the only storage player we know of that *came to market* with VM-centricity as the central point of differentiation. And they did that four years ago when VVols was a twinkle in an engineer's eye. This was a feat of innovation that was non-trivial by any standard. Today Tintri stands tall knowing that their idea of VM-centricity was right on the money and the market has endorsed it completely.

The question for the buyer is "Should you buy Tintri, knowing that VVols are coming to some traditional arrays, including the one they are likely buying today?" If the buyer wants full benefits of VM-centric storage immediately, we believe their choice is simple: Buy Tintri VMstore today and enjoy the full benefits that VM-centric storage has to offer, but do so not just with vSphere but also with Hyper-V, RHEV and others. And when VVols have matured and all the elements are functioning together enjoy support for VVols with Tintri VM-aware storage that extends to the lowest levels of the architecture.

87 Elm Street, Suite 900 · Hopkinton, MA 01748 · T: 508.435.2556 · F: 508.435.2557 www.tanejagroup.com