Monitoring VMware Virtual SAN with Virtual SAN Observer

August 2014 Edition

TECHNICAL MARKETING DOCUMENTATION

Table of Contents

Introduction

The VMware Virtual SAN Observer is designed to capture performance statistics for a VMware Virtual SAN Cluster and provide access via a web browser and capture the statistics for customer use or for VMware Technical Support.

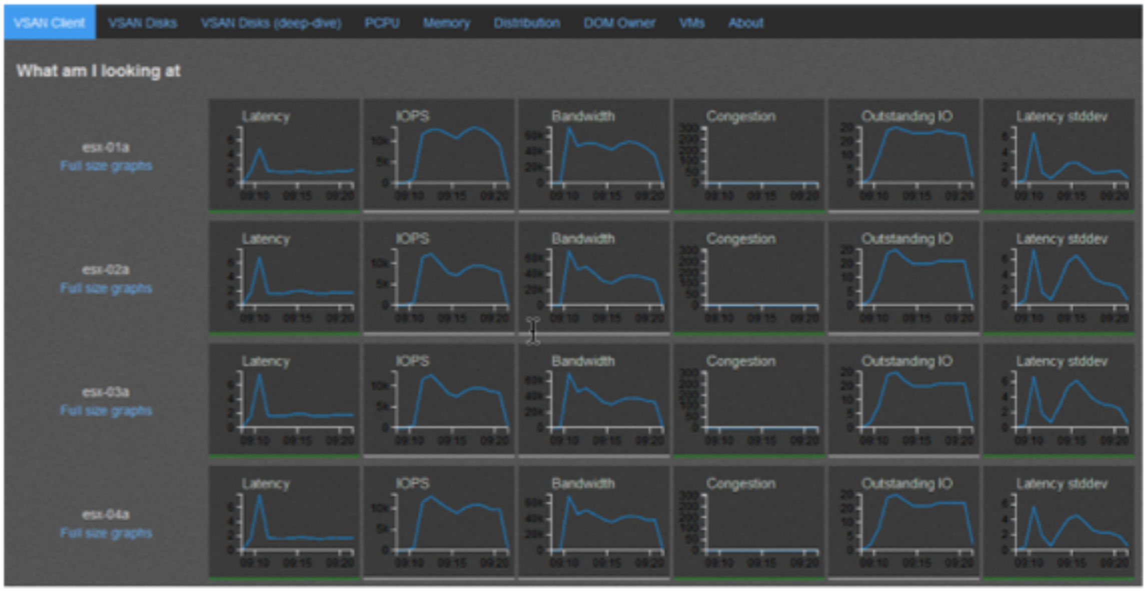
For any vSphere, Virtual SAN, and storage administrator wanting to dig deeper and analyze performance issues, the virtual SAN observer is a valuable tool.

It provides an in-depth snapshot of IOPS, latencies at different layers of Virtual SAN, read cache hits and misses, outstanding I/Os, congestion, etc. This information is provided at different layers in the Virtual SAN stack to help troubleshoot storage performance.

The Virtual SAN Observer is packaged with vSphere 5.5U1 vCenter Server. The Virtual SAN observer is part of the Ruby vSphere Console (RVC), an interactive command line shell for vSphere management that is part of both Windows and Linux vCenter Server in vSphere 5.5U1.

Initially Virtual SAN engineering team exclusively used the Virtual SAN Observer for early internal Virtual SAN troubleshooting, but the utility is now available to all VMware customers using the new vSphere 5.5U1.

Figure 1: Virtual SAN Observer UI



Starting Virtual SAN Observer

Log in to the Ruby vSphere Console (RVC) on your vCenter Server Appliance via SSH and run the command.

- rvc username@localhost

For vCenter server running on Windows, you need to run the following batch file to launch RVC

- %PROGRAMFILES%\VMware\Infrastructure\VirtualCenter Server\support\rvc\rvc.bat
- Enter the password for your vCenter Server
- Use the cd command to navigate to your vCenter Server directory, for example, localhost:
- cd localhost

Use the cd command to navigate to the data center for your Virtual SAN environment, for example, Virtual SAN:

- cd VSAN

Virtual SAN Observer Modes

We can start Virtual SAN Observer in 3 different modes.

Live monitoring for a Virtual SAN cluster named VSAN

- vsan.observer ~/computers/VSAN --run-webserver --force
- Generate a performance statistics bundle over a one hour period at 30 second intervals and save the generated statistics bundle to the /tmp folder, run the command:
- If we do not specify --max-runtime option it will by default run for 2 hours
- vsan.observer ~/computers/VSAN --run-webserver --force --generate-html-bundle /tmp --interval 30 --max-runtime 1

Offline monitoring

vsan.observer ~/computers/VSAN --generate-html-bundle /tmp

- Builds a tar.gz bundle with HTML files and can be sent via email to colleagues/VMware technical support

Full raw stats bundle

vsan.observer ~/computers/VSAN --filename /tmp/vsan-observer-file-sample.json

- Large json file with all observed metrics
- Ideal for VMware support/engineering to perform deeper analysis

The vCenter Server retains the entire history of the observer session in memory until it is stopped with Ctrl+C.

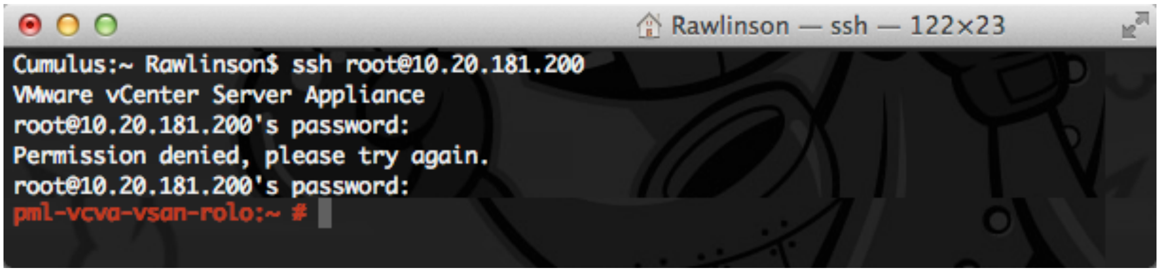
We can view live statistics when the command is running, navigate to your vCenter Server and specified port number using the URL:

- http://vCenterServer_hostname_or_IP_Address:8010

Enabling Virtual SAN

Log into vCenter Server Virtual Appliance (VCSA), Use a terminal program to open an SSH connection to the

Figure 2: SSH to VCVA



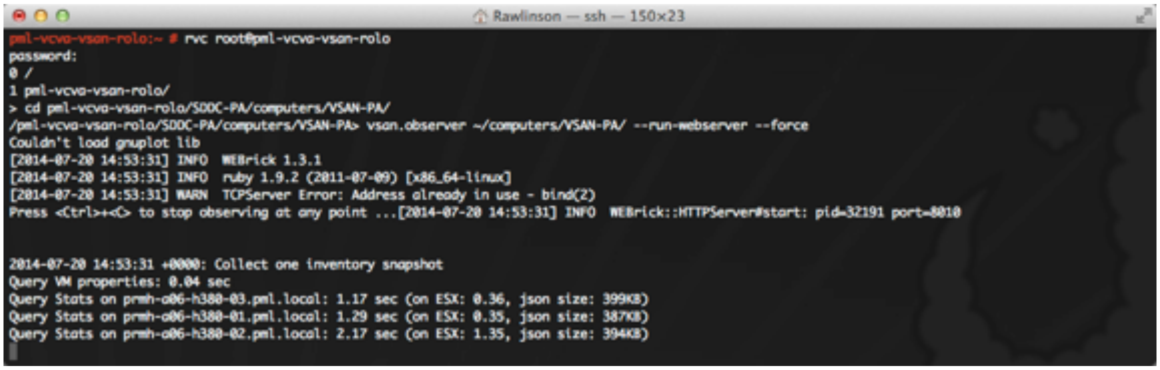
```
Cumulus:~ Rawlinson$ ssh root@10.20.181.200
VMware vCenter Server Appliance
root@10.20.181.200's password:
Permission denied, please try again.
root@10.20.181.200's password:
pml-vcva-vsan-rolo:~ #
```

Enabling Virtual SAN Web Client is a prerequisite step to view the graphs in the Observer tool

Starting Virtual SAN Observer Live

After logging into vCenter Server
1. Type "rvc user@<ip or hostname>"
2. Enter password
3. Type cd localhost/Datacenter/computers/<cluster name>
4. Enter vsan.observer ~/computers/<cluster name>/ --run-webserver –force

Figure 3: Starting the Virtual SAN Observer



```
pml-vcva-vsan-rolo:~ # rvc root@pml-vcva-vsan-rolo
password:
0 /
1 pml-vcva-vsan-rolo/
> cd pml-vcva-vsan-rolo/SDDC-PA/computers/VSAN-PA/
/pml-vcva-vsan-rolo/SDDC-PA/computers/VSAN-PA> vsan.observer ~/computers/VSAN-PA/ --run-webserver --force
Couldn't load gnuplot lib
[2014-07-20 14:53:31] INFO  WEBrick 1.3.1
[2014-07-20 14:53:31] INFO  ruby 1.9.2 (2011-07-09) [x86_64-linux]
[2014-07-20 14:53:31] WARN  TCPServer Error: Address already in use - bind(2)
Press <Ctrl>+<C> to stop observing at any point ...[2014-07-20 14:53:31] INFO  WEBrick::HTTPServer#start: pid=32191 port=8010


2014-07-20 14:53:31 +0000: Collect one inventory snapshot
Query VM properties: 0.04 sec
Query Stats on prmh-a06-h380-03.pml.local: 1.17 sec (on ESX: 0.36, json size: 399KB)
Query Stats on prmh-a06-h380-01.pml.local: 1.29 sec (on ESX: 0.35, json size: 387KB)
Query Stats on prmh-a06-h380-02.pml.local: 2.17 sec (on ESX: 1.35, json size: 394KB)
```

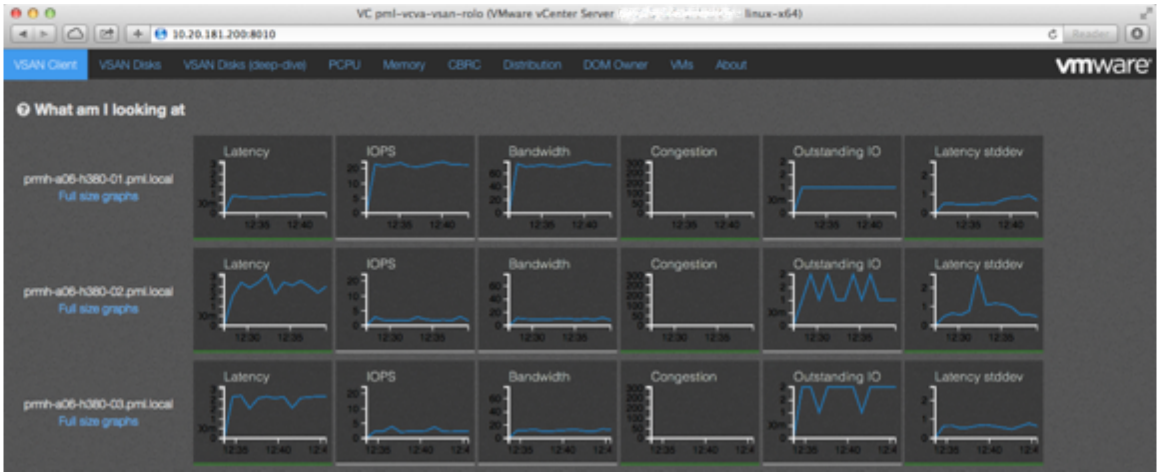
The above command will start Virtual SAN Observer. In a typical customer Virtual SAN deployment setup the default number given after --max- runtime is counted in terms of hours.

View Virtual SAN Observer In Browser

Open a web browser and type the URL in the address bar based on the server IP or FQDN

- http://<VCVA IP or FQDN>:8010/

Figure 4: Accessing Virtual SAN Observer



In the following chapters, we will dig deeper into most of these tabs in the Observer UI. The Virtual SAN Observer default port number is 8010.

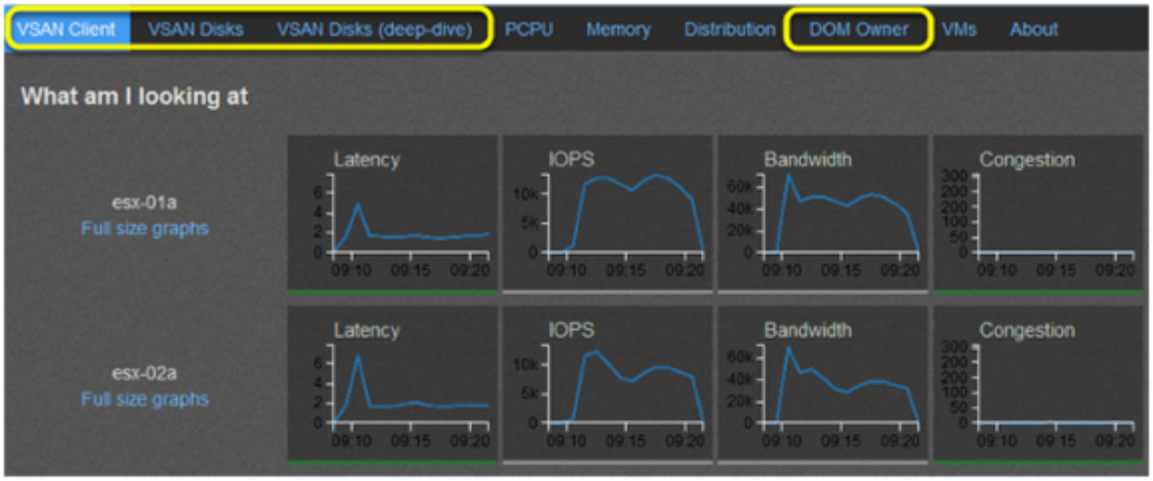
Observer UI Walk-through

We will now walk through the available tabs in Virtual SAN observer UI and explain what information each tab holds.

VSAN Disk Tabs

When we connect to Virtual SAN observer using a browser we see the VSAN Client page as show above. There are four tabs dedicated to per host disk performance counters.

Figure 5: Virtual SAN Observer Disk Tabs



VSAN Client: Each host in the Virtual SAN cluster runs a VSAN client. This is the upper most layer of Virtual SAN. The graphs displayed in this page provide performance statistics as perceived by Virtual Machines in the respective hosts. This presents the first step to troubleshooting Virtual SAN performance issues.

VSAN Disks: Beneath the VSAN Client layer is the VSAN Disks layer. This layer is responsible for all the physical disks in each host. As a result VSAN Disks tab shows statistics related to physical disks in each host. This tab provides a reasonable insight into the performance of physical disks in every host in the cluster.

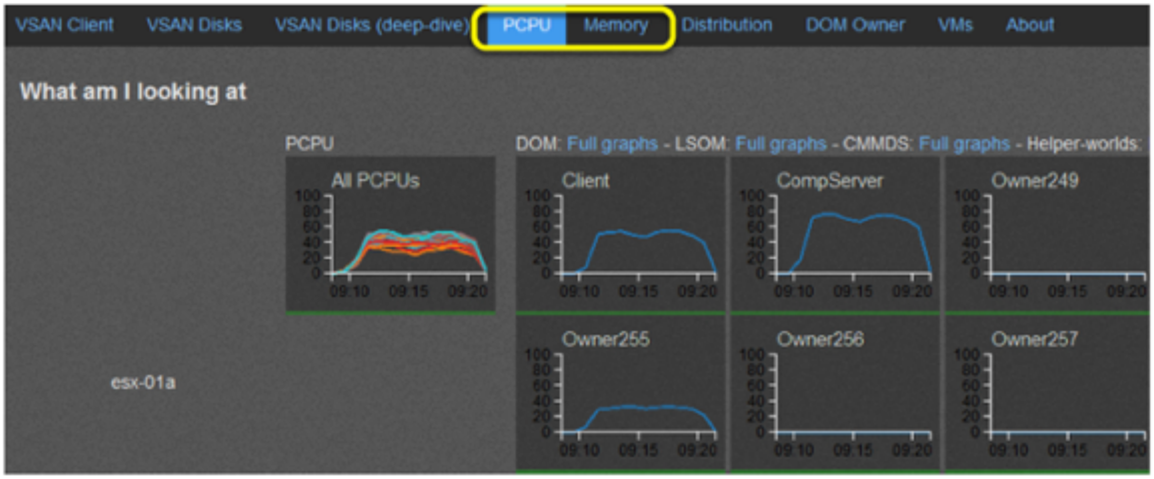
VSAN Disks (deep-dive): This tab provides a wealth of information broken down to each individual physical disks (SSD, HDD). It provides read cache hit rates, cache evictions, and write buffer fill for SSD. Also, latency and IOPS stats are provided for all disks in each node.

DOM Owner: This layer implements RAID and Recovery and is responsible for originating resync and recovery operations. Usually runs on the same host as VSAN Client. This tab is mainly for use by VMware GSS and Virtual SAN developers to look at.

CPU & Memory Tabs

The PCPU and Memory tabs provide per-host CPU and memory statistics.

Figure 6: CPU & Memory Tabs



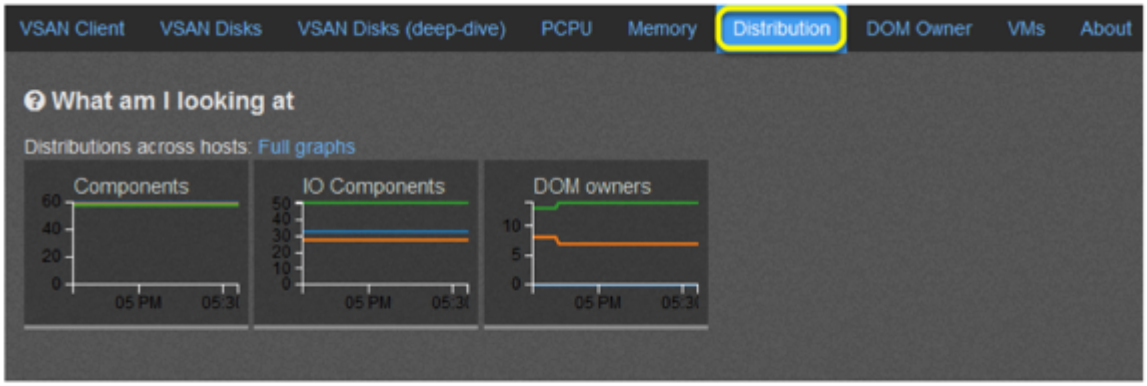
PCPU: This tab shows overall CPU usage and also per component CPU usage statistics of Virtual SAN. It also shows CPU usage of Virtual SAN networking components. Extended high CPU usage may result in storage performance issues.

Memory: Shows memory consumption of various Virtual SAN components. Usually these are fully allocated and used. This is mainly meant for VMware technical support personnel.

Distribution Tab

This tab shows how well Virtual SAN is balancing the objects (VMDKs, delta disks, witness) and components across hosts in the cluster. Each object can be broken down into components and the components themselves are distributed across hosts in the cluster. Each host has a 3000 component limit.

Figure 7: Distribution Tab



What am I looking at

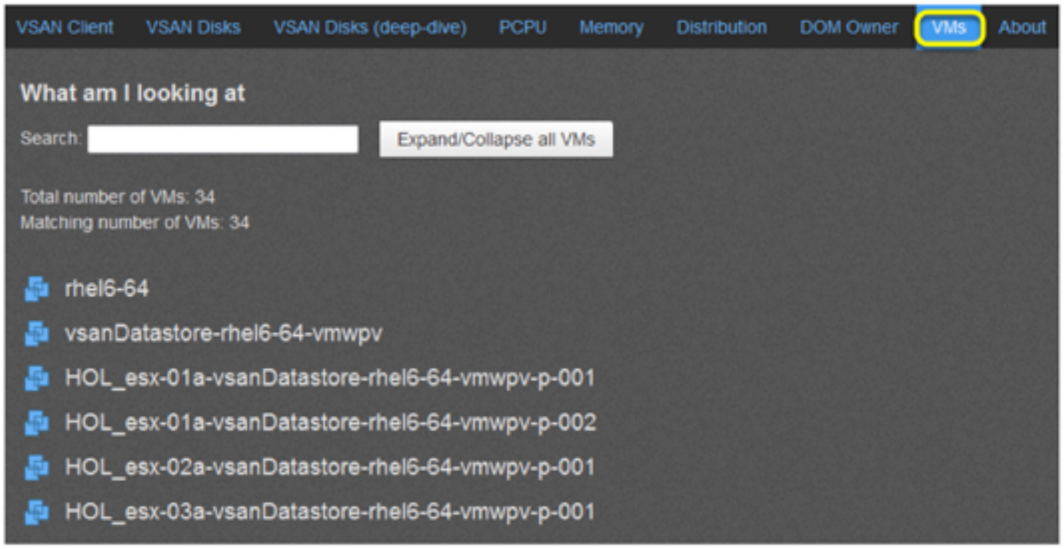
Distributions across hosts: Full graphs

VMs Tab

Each VM has a directory that holds non-vmdk files that contain files like VM configuration and VM log files. In addition to this each VM can have one or more virtual disks. Each of these virtual disks represents a single entity.

The presence of snapshots will transform each virtual disk into multiple backing entities. This tab provides storage performance statistics as seen by each VM. Latency, IOPS, read cache hit rates and eviction statistics are provided at the VM directory, virtual disk and backing disks level. Drilling down at a VM level is conveniently fulfilled by this tab.

Figure 8: VMs Tab



What am I looking at

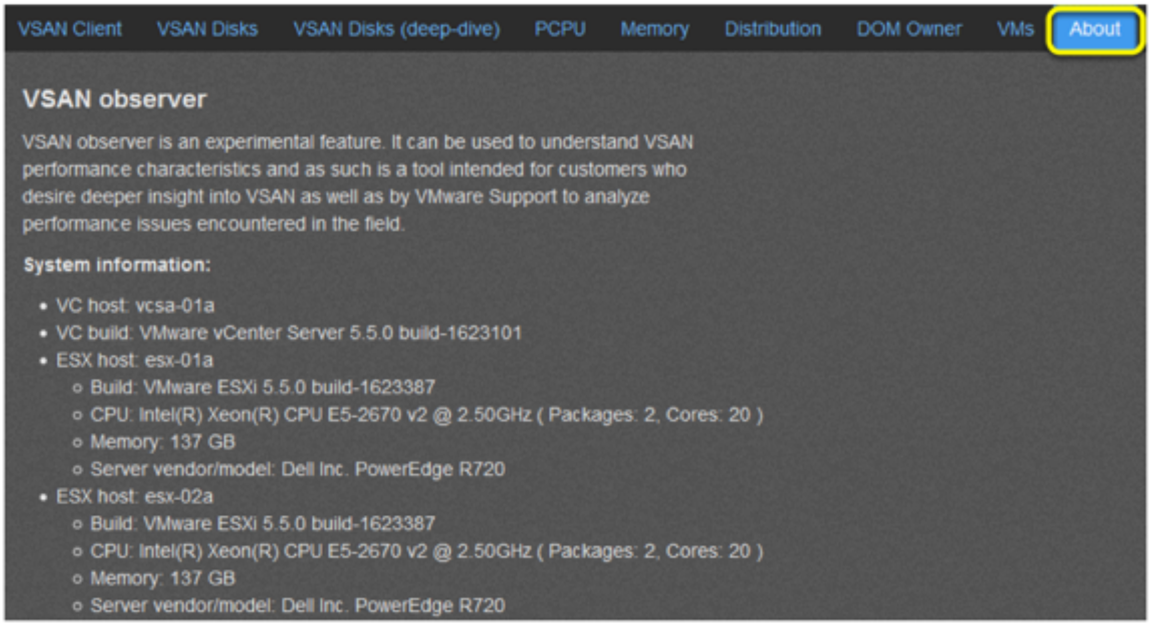
Total number of VMs: 34
Matching number of VMs: 34

rhel6-64
vsanDatastore-rhel6-64-vmwpv
HOL_esx-01a-vsanDatastore-rhel6-64-vmwpv-p-001
HOL_esx-01a-vsanDatastore-rhel6-64-vmwpv-p-002
HOL_esx-02a-vsanDatastore-rhel6-64-vmwpv-p-001
HOL_esx-03a-vsanDatastore-rhel6-64-vmwpv-p-001

About Tab

The "About" tab provides various hardware and software information about ESX hosts in the cluster. We can get a quick glimpse of disks, memory, CPU, and software information on each ESX host in the cluster.

Figure 9: About Tab



VSAN observer

VSAN observer is an experimental feature. It can be used to understand VSAN performance characteristics and as such is a tool intended for customers who desire deeper insight into VSAN as well as by VMware Support to analyze performance issues encountered in the field.

System information:

- VC host: vcsa-01a
- VC build: VMware vCenter Server 5.5.0 build-1623101
- ESX host: esx-01a
  - Build: VMware ESXi 5.5.0 build-1623387
  - CPU: Intel(R) Xeon(R) CPU E5-2670 v2 @ 2.50GHz (Packages: 2, Cores: 20)
  - Memory: 137 GB
  - Server vendor/model: Dell Inc. PowerEdge R720
- ESX host: esx-02a
  - Build: VMware ESXi 5.5.0 build-1623387
  - CPU: Intel(R) Xeon(R) CPU E5-2670 v2 @ 2.50GHz (Packages: 2, Cores: 20)
  - Memory: 137 GB
  - Server vendor/model: Dell Inc. PowerEdge R720

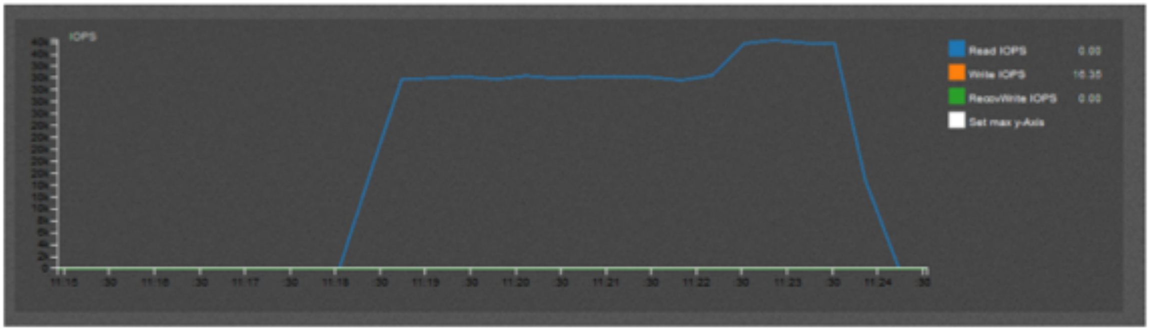
Understanding Virtual SAN Storage Performance

In this chapter we will understand some of the key concepts and terminologies pertaining to storage performance as applicable to virtual SAN

IOPS

IOPS gives a measure of number of Input/Output Operations Per Second of a storage system. An I/O operation is typically a read or a write and a size. I/O size can vary from anywhere between a few bytes and several megabytes.

Figure 10: IOPS Graph



If we see high IOPS in Virtual SAN it does not mean that we have a problem, all it means that we are using the storage to its maximum. A disk clone or VM clone operation would use all available IOPS thereby completing the operation in least possible time.

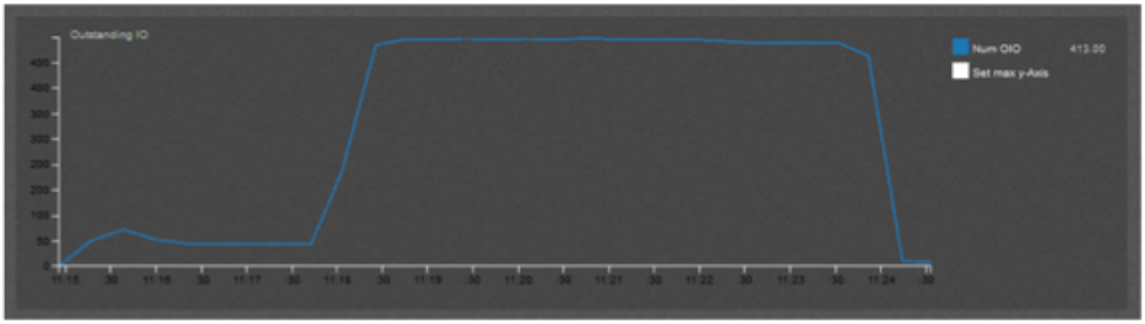
Low IOPS may also not mean that there is an immediate problem; it could simply be that the I/O sizes are large. Large I/O sizes could lead to lower IOPS.

Outstanding I/O

When a virtual machine requests for certain IO to be performed (reads or writes), these requests are sent to storage devices. Until these requests are complete they are termed outstanding I/Os.

Large number of outstanding I/Os can have an adverse effect on the device latency. Storage controllers that have a large queue depth can handle higher outstanding IOs.

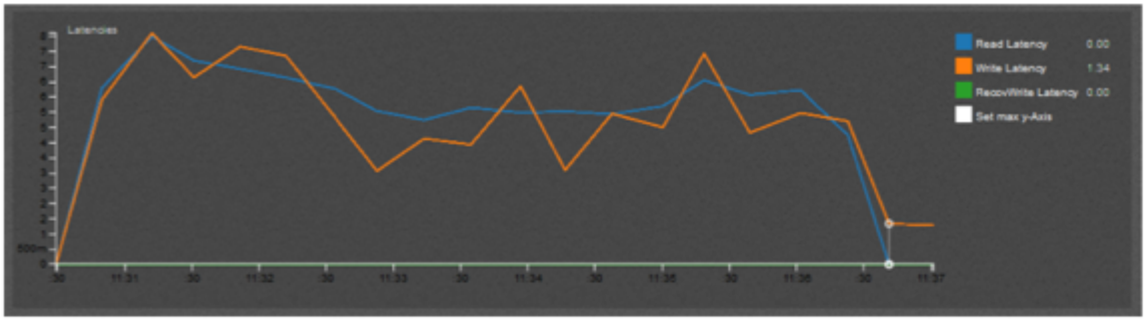
Figure 11: Outstanding I/O Graphs



Latency

Latency gives a measure of how long it takes to complete one I/O operation from an application's viewpoint. As we know that I/O sizes can vary from a few bytes to several megabytes it follows that latency can vary based on the size of the I/O.

Figure 12: Latency Graphs



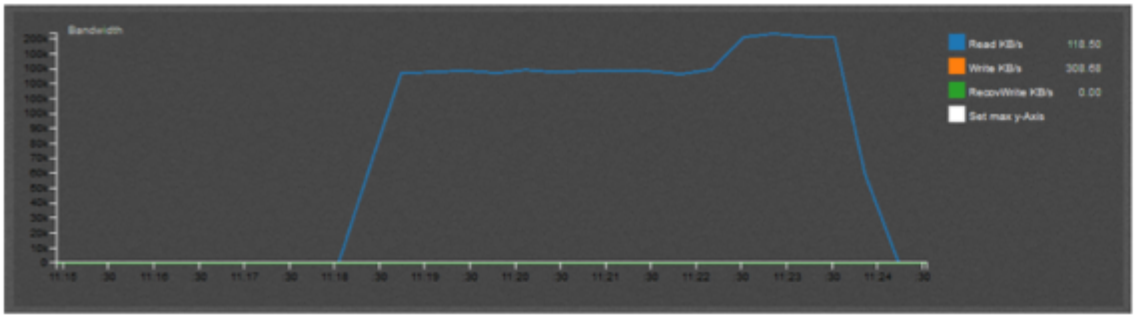
Virtual SAN uses flash based devices (SSD) to reduce the effective latency as seen by a virtual machine. If we see relatively high latency in Virtual SAN it could mean on of the following:

- Large average I/O sizes, which leads to increased latencies
- I/Os are predominantly writes. SSDs are faster at reads than writes.
- Large number of outstanding I/Os or just too many VMs busy doing large number of I/Os
- Slow SSD that is simply hard pressed to keep up with I/Os coming in
- Too many random reads causing read cache misses in the SSD. A large number of random read requests from a VM can result in I/Os being served from the underlying magnetic disks rather than from SSD cache.

Bandwidth

Bandwidth measures the data rate that storage is capable of. Now is a good time to understand how IOPS and bandwidth may influence.

Figure 13: Bandwidth Graph



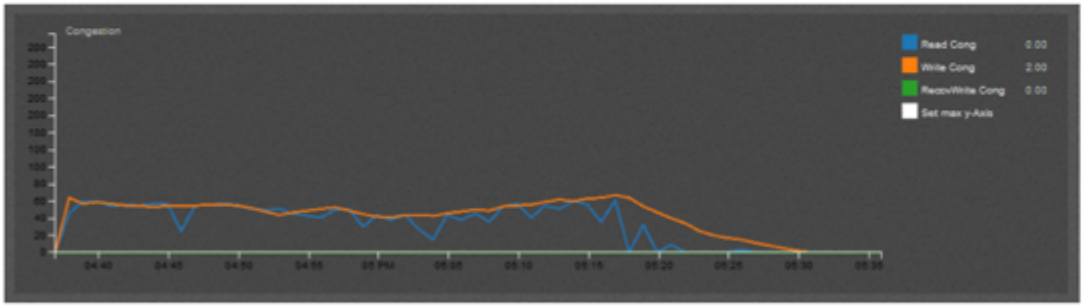
- When small I/O sizes are involved a device may hit the maximum IOPS ceiling before exhausting the available bandwidth provided by the device, or controller, or the underlying physical link.
- Conversely, for large I/Os the bandwidth may become a limiting factor before the maximum IOPS of a device or controller

When troubleshooting storage performance we have to look at IOPS, I/O sizes, outstanding I/Os, and latency to get a complete picture.

Congestion

Congestion in Virtual SAN happens when typically lower layers fail to keep up with the I/O rate of higher layers. For example if VMs are performing a lot of write operations it could lead to filling up of write buffers. These buffers have to be de-staged to magnetic disks.

Figure 14: Congestion



However, they can only be done at a slower rate than SSDs. This causes Virtual SAN to artificially introduce latencies in the VMs in order to slow down writes so that write buffers can be freed up. Congestion is not normal and in most cases congestion will be close to zero.

Putting It All Together

We will put all the concepts discussed earlier into an example. Consider a typical mid range SATA SSD with the following characteristics.

- 50000 read IOPS at 4k size
- 40000 write IOPS at 4k size
- Maximum IOPS is at 4 Outstanding IOs (OIO)

Write latency can be calculated as:

- Write Latency = OIO/(Write IOPS) = 4/40000 = 0.1ms

Read latency can be calculated with the same formula. We leave it as an exercise for the user to calculate read latency.

If OIO is increased to 16 the latency increases. We can find that by using the same formula

- Write Latency = 16/40000 = 0.4ms

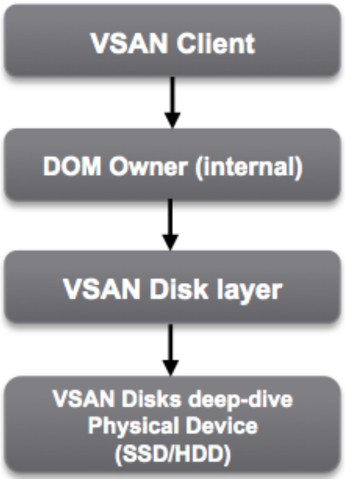
Increasing I/O size can roughly increase latency proportionally. There are exceptions to this rule, however, for simplicity we will take this as a rough guideline.

- Write Latency (at 8k I/O and 4 OIO) = 2x the latency of 4k Write I/O = 2 * 0.1 = 0.2ms
- Similarly write IOPS at 8k reduces by half ~ 20000 IOPS

Virtual SAN Architecture

There are four main high-level components in Virtual SAN

Figure 15: Virtual SAN Observer Monitoring Architecture



VSAN Client

- This is the client that provides access to Virtual SAN storage. It runs on all hosts that are running VMs and performs I/O on behalf of the VMs.

DOM Owner

- Usually runs on the same host as VSAN Client.
- This layer implements RAID and Recovery and is responsible for originating resync and recovery operations.

VSAN Disks layer

- This is the layer that actually serves IO from local disks to different nodes in the cluster.
- Data that is needed by a VM may not necessarily reside in the same host as the VM. In such cases Virtual SAN disks layer on the node where the data resides serves the data over the network to the requesting node
- This layer also has a fair scheduler that may queue IOs that are coming in

Disk deep-dive/physical devices

- This is the layer that actually does the IO to SSD and/or HDD

These four main blocks are directly mapped in the Virtual SAN observer UI to aid in better analysis of performance statistics at different layers of Virtual SAN

Virtual SAN Observer Analysis Sample I

In this chapter we will try to analyze a 4k 80% read workload and understand the behavior and impact at different layers.

4k IO Size (80% Read, 4 Outstanding IOs)

In this example we have a workload running small (4096 bytes) I/O that are predominantly reads (80%) with 4 outstanding I/Os per VM at any point.

- We will show the various tabs in the UI corresponding to this workload
- Optionally, we invite users to delve deeper by clicking on the full graphs wherever
- applicable for better understanding.
- Please click on the shortcut named "Analysis_1.html" on the desktop
- We request you to follow our analysis on the browser by navigating to the same tab as the subsequent steps

VSAN Client Tab

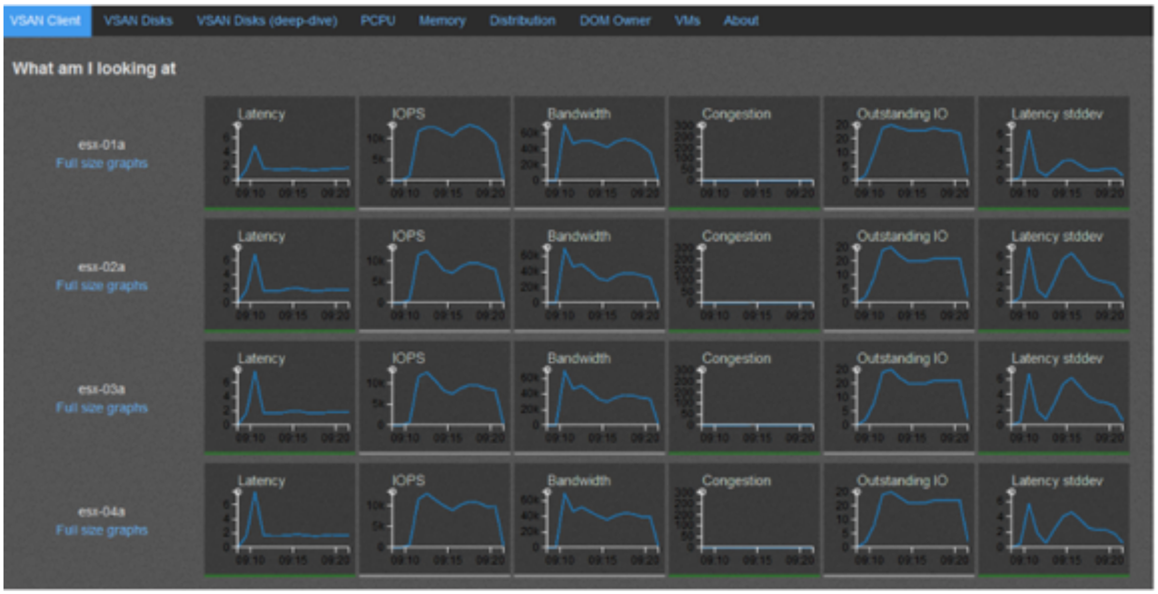


Figure 16: Virtual SAN Client Tab

- We have 4 hosts in the cluster
- Each node has 8 virtual machines generating I/O with each virtual machine having 4 outstanding I/Os

We will now walk through each graph in this screenshot

Latency

- Initially we see a short spike in latency. This is due to random read operations that happened for the first time on different blocks. As a result it generated some cache misses and the blocks had to be fetched from the magnetic disks rather than from the SSDs.
- Eventually latency drops to < 2ms range indicating that all subsequent random reads were successfully hitting the cache.
- This pattern holds true for all 4 hosts as expected

- Notice all the latency graphs are underlined green, indicating that the latency is well under 30ms threshold for more than 70% of I/Os

IOPS

IOPS graph shows no anomalies. It picks up as soon as latency drops and stays steady at around 12000 IOPS

Outstanding IO

- We know that each VM had 4 outstanding IOs
- Since each host has 8 VMs running, the outstanding IO at the layer increases and settles at around 20
- OIO need not equal the total OIO across all VMs

Latency Standard Deviation

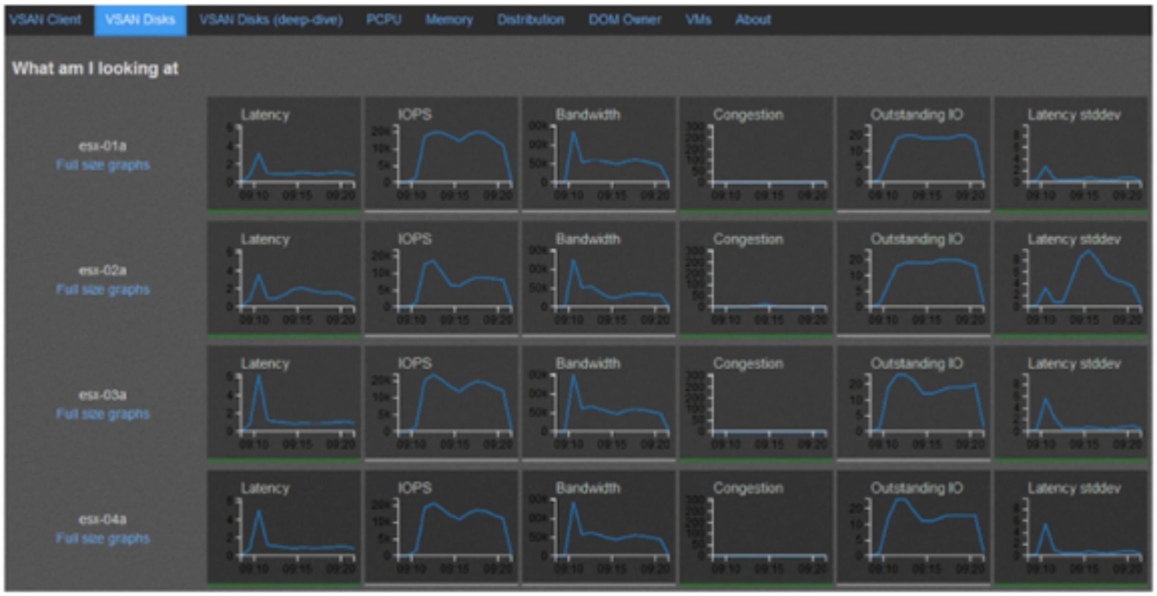
- The latency stddev graph is included to give an idea of how wide the latency spread is at any given point
- Lower standard deviation implies that the latency is predictable and well bounded
- Higher values indicate that latency is fluctuating heavily. This could very well be due to impacts from lower layers

Under each of the hosts there is a link to full size graphs of what is visible in the thumbnail. We invite the user to click on the Analysis_1.html link on the desktop, included as part of the lab, and open in a browser to explore the full size graphs.

VSAN Disks Tab

We know from a previous chapter that VSAN disks layer is responsible for servicing the IO from local disks and may be residing on different nodes in the cluster. We will analyze this layer from the graphs presented by Virtual SAN observer.

Figure 17: Virtual SAN Disk Tab



Latency & IOPS

- The latency curve in this example is well under threshold. It follows very closely what we saw in the earlier graph. Latency spikes in the beginning and then settles down low.
- IOPS also corresponds to what was happening in the VSAN Client layer as we say in the earlier graphs.
- IOPS is much higher than what the earlier layer reported. This is due to the presence of additional IO corresponding to RAID and sync operations at this layer

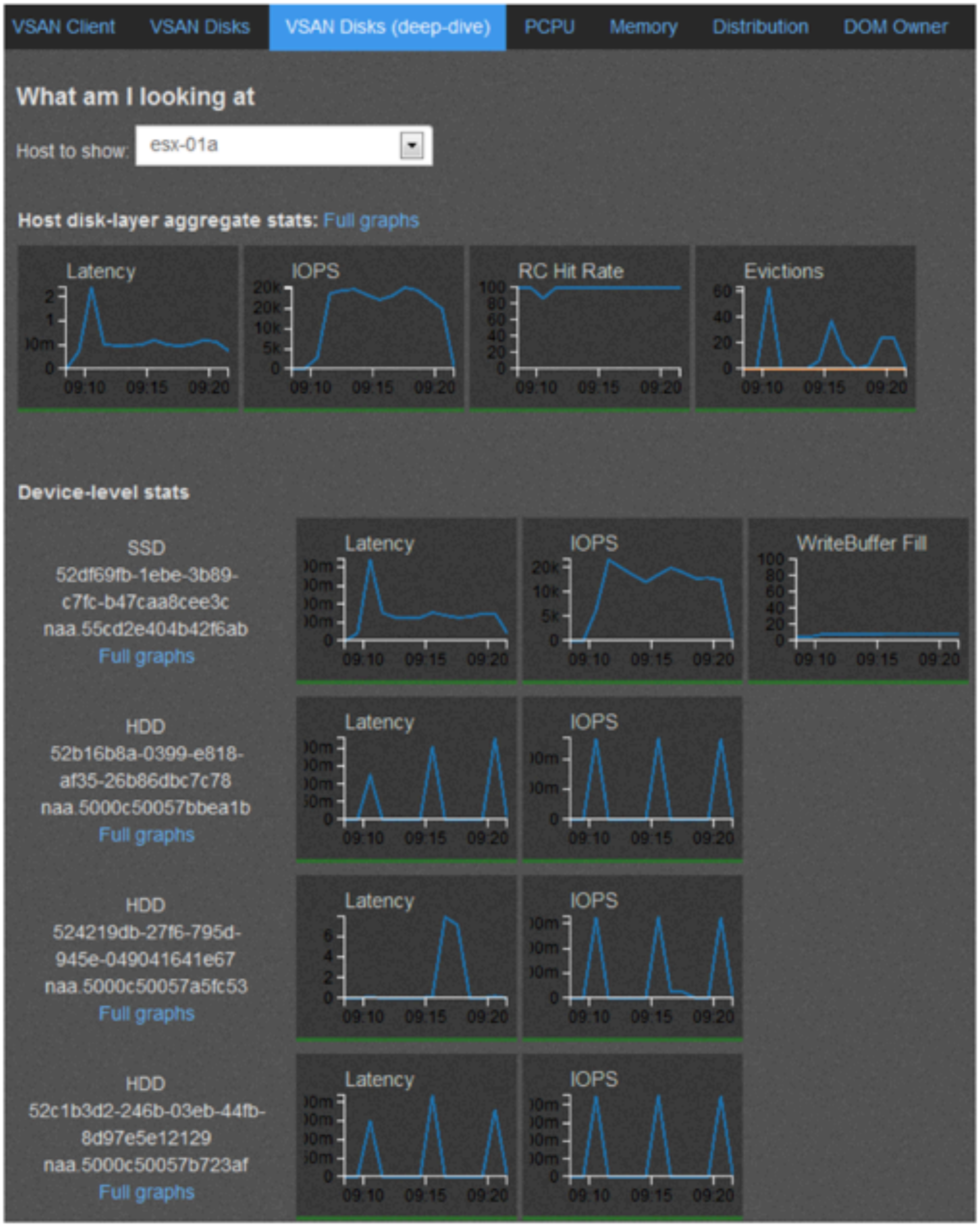
Outstanding IO

- Outstanding IO is almost the same across all hosts, indicating components are well balanced across hosts.

In general this host is performing well under this workload with all parameters well under threshold. Next, let us look at the physical disk layer.

VSAN Disks (Deep-Dive) Tab

Figure 18: Virtual SAN Disks (Deep Dive) Tab



What am I looking at

Host to show: esx-01a

Host disk-layer aggregate stats: Full graphs

Device-level stats

SSD
52df69fb-1ebe-3b89-c7fc-b47caa8cee3c
naa.55cd2e404b42f6ab
Full graphs

HDD
52b16b8a-0399-e818-af35-26b86dbc7c78
naa.5000c50057bbea1b
Full graphs

HDD
524219db-27f6-795d-945e-049041641e67
naa.5000c50057a5fc53
Full graphs

HDD
52c1b3d2-246b-03eb-44fb-8d97e5e12129
naa.5000c50057b723af
Full graphs

We are now looking at the physical disk layer of host esx-01a

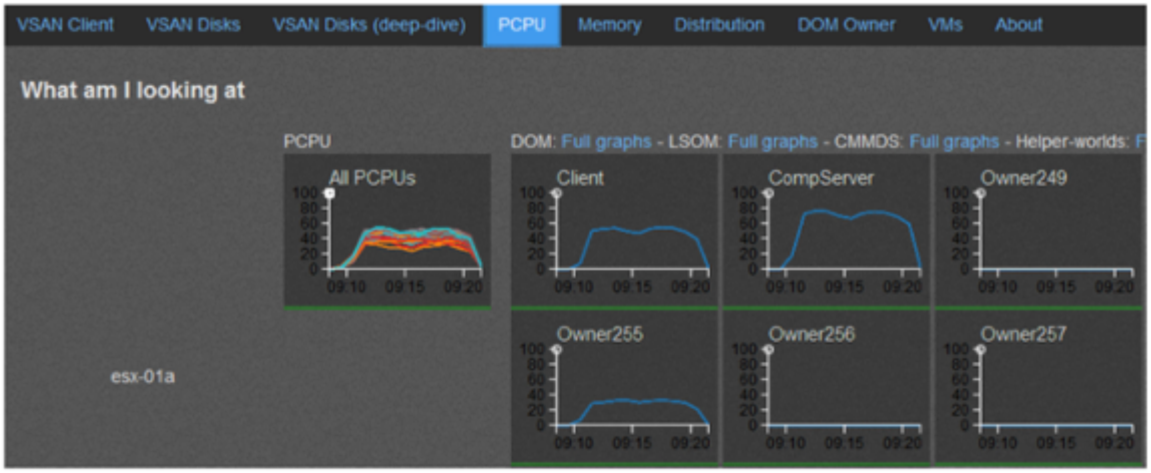
- Latency and IOPS correspond closely to what we have seen in the earlier graphs

- If we look at RC Hit rate, we can see that there is a dip in read cache hit rate. This corresponds to the initial latency spike we saw in the earlier graphs
- Device level stats shows all the disks involved in this disk group. In our example we have one SSD and 3 HDDs in the only disk group in the host
- The SSD is performing well with less than 1ms latency for the most part
- All the physical disks in this disk group are performing reasonably on par with each other.
- Write buffer is around 5% full indicating a healthy SSD to disk capacity ratio for this workload

PCPU Tab

We show a partial screen-shot of the PCPU tab encompassing a few graphs in host esx-01a. This tab shows CPU usage of VSAN client, component server (VSAN disks layer), different component owners, and LSOM (VSAN Disks deep-dive layer).

Figure 19: Virtual SAN PCPU Tab

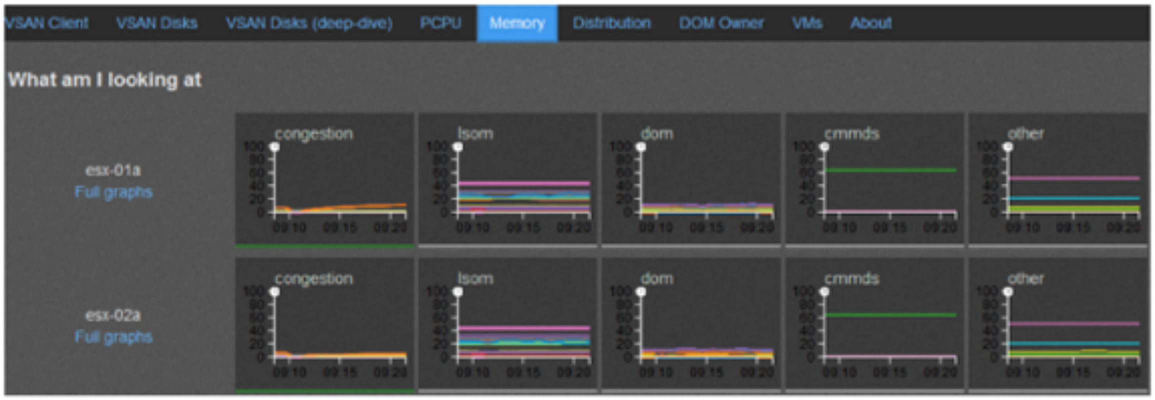


- Here we see that the client and component server using around 30 - 50% CPU during the time IO was actively generated.
- The different component owners themselves are seen using much less CPU
- Overall this layer is healthy and performing well under the generated workload

- Slow performance combined with extended high CPU consumption may point at a performance problem. However, high CPU consumption in itself may not be a sign of any issue

Memory

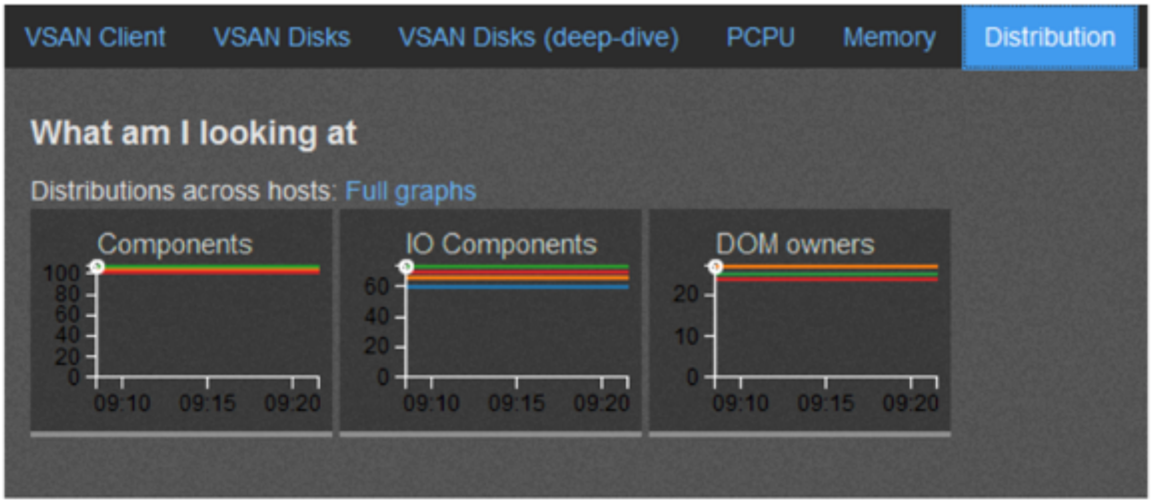
Figure 20: Virtual SAN Memory Tab



This tab shows memory consumption of different memory pools. The most important graph to look at is congestion. If memory consumption of congestion pool is very high it usually is indicative of an underlying performance problem.

Distribution Tab

Figure 21: Virtual SAN Distribution Tab

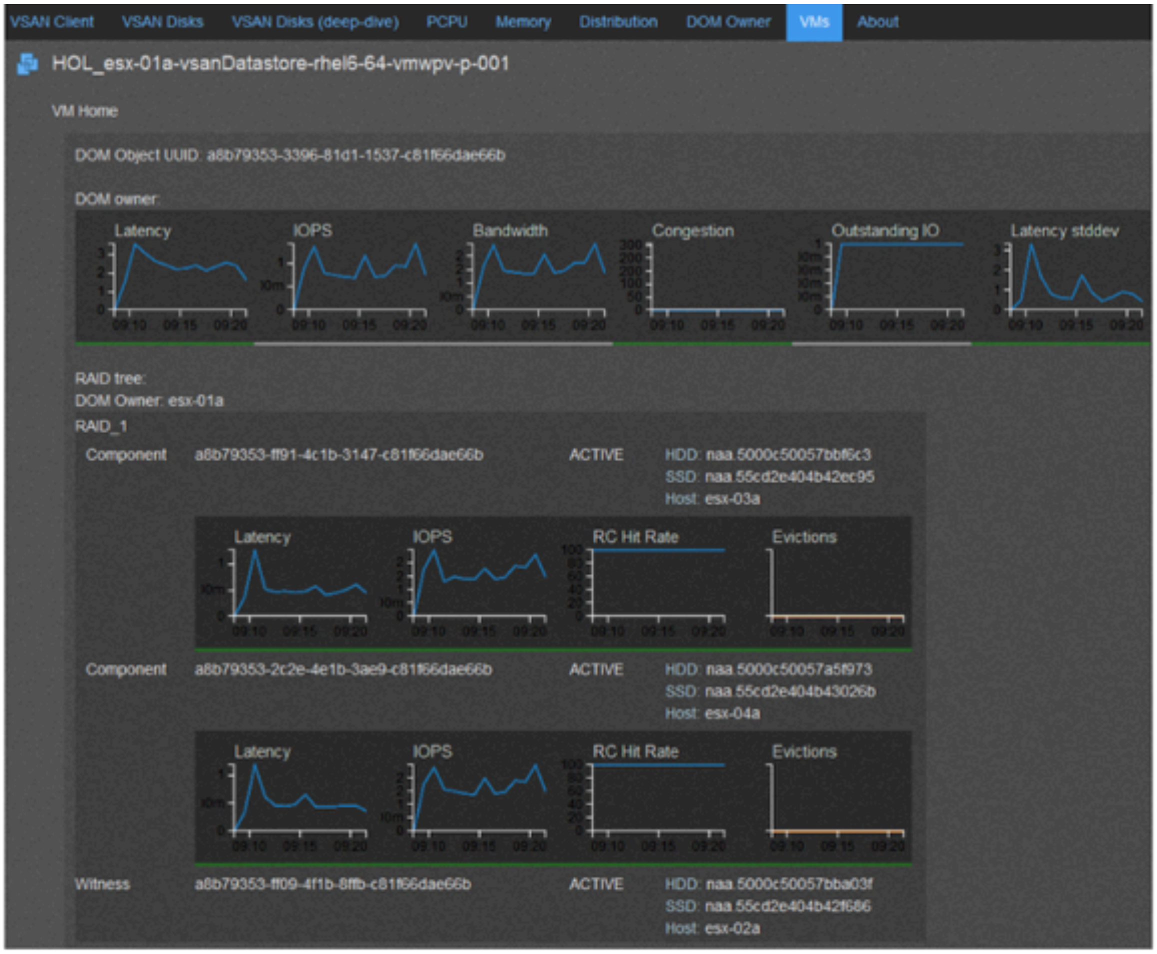


In this tab the distribution of components is shown in a graphical way. Each host has a 3000 component limit. A uniform line indicates a well balanced system. Removal of disk groups will result in a re-balancing of components, which would show up as fluctuations in these graphs.

VMs Tab

All VMs are listed under VMs tab. Selecting a particular VM provides more details about the object layout of that VM. There are two sections under each VM -- VM Home and Virtual Disk.

Figure 22: Virtual SAN VMs Tab



HOL_esx-01a-vsanDatastore-rhel6-64-vmwpv-p-001

VM Home

DOM Object UUID: a8b79353-3396-81d1-1537-c81f66dae66b

DOM owner:

RAID tree:
DOM Owner: esx-01a
RAID_1

Component a8b79353-ff91-4c1b-3147-c81f66dae66b ACTIVE HDD: naa.5000c50057bbf6c3
SSD: naa.55cd2e404b42ec95
Host: esx-03a

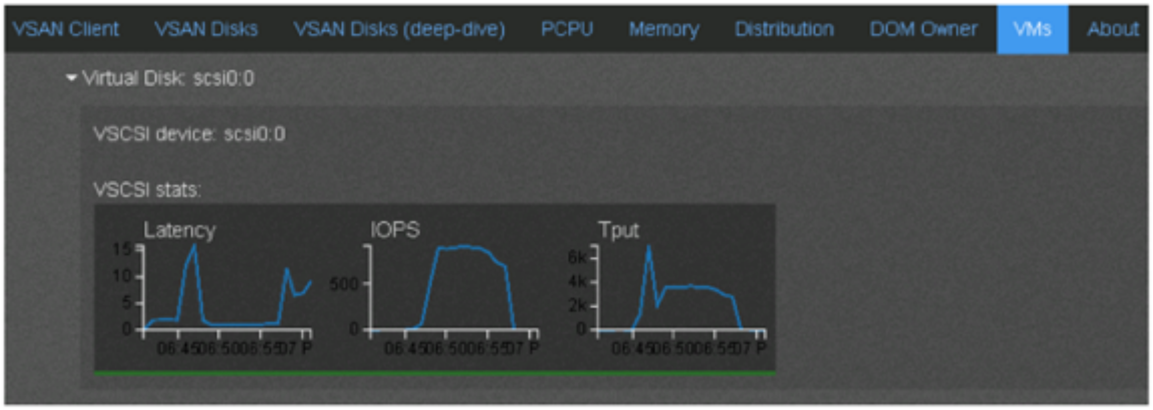
Component a8b79353-2c2e-4e1b-3ae9-c81f66dae66b ACTIVE HDD: naa.5000c50057a5f973
SSD: naa.55cd2e404b43026b
Host: esx-04a

Witness a8b79353-ff09-4f1b-8ffb-c81f66dae66b ACTIVE HDD: naa.5000c50057bba03f
SSD: naa.55cd2e404b42f686
Host: esx-02a

VM Home

This is where the VM's configuration file, its log files, and other VM related small files are stored. The RAID tree subsection shows the different component owners for VM Home.

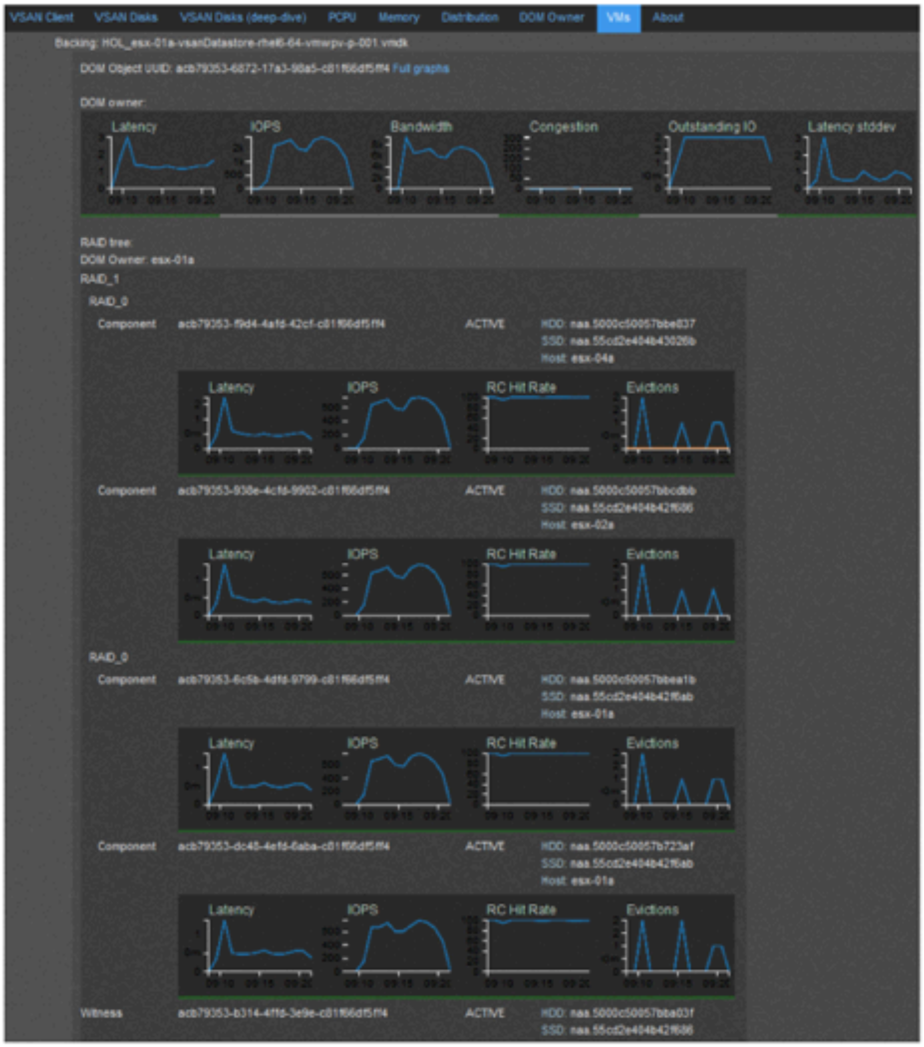
Figure 23: Virtual SAN VMs Tab



We now turn our attention to the second main section under each VM.

Virtual Disk

Figure 24: Virtual SAN VMs Expanded View



Backing: HOL_esx-01a-vsanDatastore-rhel6-64-vmwpv-p-001.vmdk

DOM Object UUID: acb79353-6872-17a3-98a5-c81f66df5ff4

DOM owner:

RAID tree:
DOM Owner: esx-01a
RAID_1
RAID_0

Component	acb79353-f9d4-4afd-42cf-c81f66df5ff4	ACTIVE	HDD: naa.5000c50057bbe837 SSD: naa.55cd2e404b43026b Host: esx-04a
Component	acb79353-938e-4cfd-9902-c81f66df5ff4	ACTIVE	HDD: naa.5000c50057bbcdbb SSD: naa.55cd2e404b42f686 Host: esx-02a

RAID_0

Component	acb79353-6c5b-4dfd-9799-c81f66df5ff4	ACTIVE	HDD: naa.5000c50057bbea1b SSD: naa.55cd2e404b42f6ab Host: esx-01a
Component	acb79353-dc48-4efd-6aba-c81f66df5ff4	ACTIVE	HDD: naa.5000c50057b723af SSD: naa.55cd2e404b42f6ab Host: esx-01a
Witness	acb79353-b314-4ff6-3e9e-c81f66df5ff4	ACTIVE	HDD: naa.5000c50057bba03f SSD: naa.55cd2e404b42f686

- This shows the different virtual disks attached to the VM. Each disk displays stats specific to that disk. We can drill down performance stats to the individual virtual disks
- The VSCSI layer shows the aggregate latency, IOPS and throughput numbers for a particular virtual disk of a VM.

This expanded view of backing disk shows a lot of information.

- First off the DOM owner shows the same aggregate as the VSCSI layer shown in the previous screenshot.

- Here we see that the both the DOM owner and the VSCSI stats from the previous graph shown about 2000 IOPS aggregate at this layer.

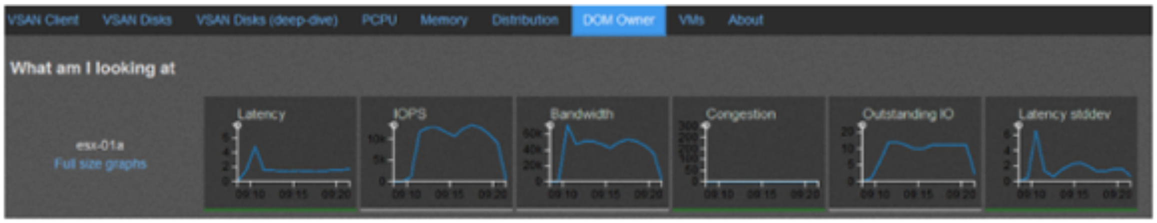
Moving further down we come to the RAID tree view.

- In our set up the RAID tree has RAID 1 layout with components distributed across different SSDs/ HDDs and hosts in the cluster.
- Failure to Tolerate (FTT) is set to 1. So we have 2 replicas for each component
- We can also see the different RAID0 (stripes) components under each mirror since the set up has a stripe width of 2
- Each component lists the active SSDs, HDDs and the host that they reside on.
- Presence of stripe and mirror helps improving both read and write/destage performance at the HDD disks layer
- In this example we can see that the aggregate IOPS is indeed equal to almost the sum of all component IOPS.
- In addition to RAID/stripe components we see something called a "Witness"
- Witness is a metadata disk. If the replicas of the VMDKs are placed on two different hosts, the witness is placed on a third host. This means that if any
- single host fails, we still have a copy of the data, and we still have greater than 50% of the components available to satisfy the availability requirement of an object

DOM Owner

We explained earlier that every object in Virtual SAN has an owner and that it is responsible for providing RAID and resync services to ensure correctness. Virtual SAN tries to co-locate the owner and the client to not incur an additional network hop.

Figure 25: DOM Owner



The main purpose of this tab is for the use of VMware GSS and Virtual SAN developers. We do not recommend users delving into the full graphs of DOM owner tab as advanced analysis and manual correlation needs to be performed in order to understand data in this layer.

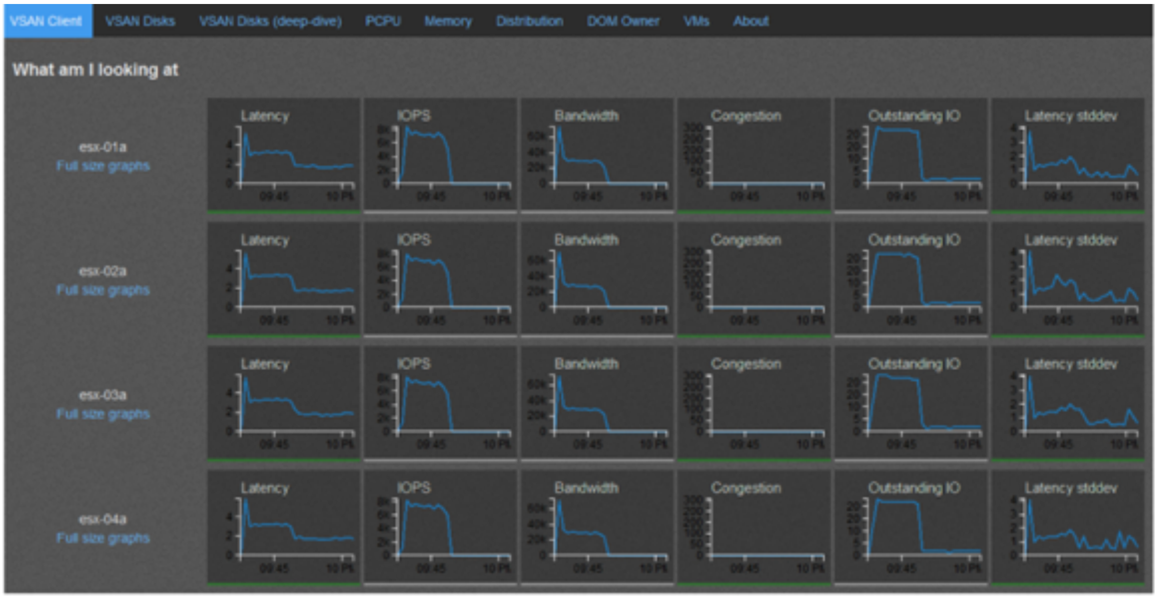
Virtual SAN Observer Analysis Sample II

We will continue our analysis in the chapter but with a write intensive workload as opposed to a read intensive one we analyzed in the last chapter.

4k IO Size (80% Write, 4 Outstanding IO)

VSAN Client Tab

Figure 26: Virtual SAN Client Tab

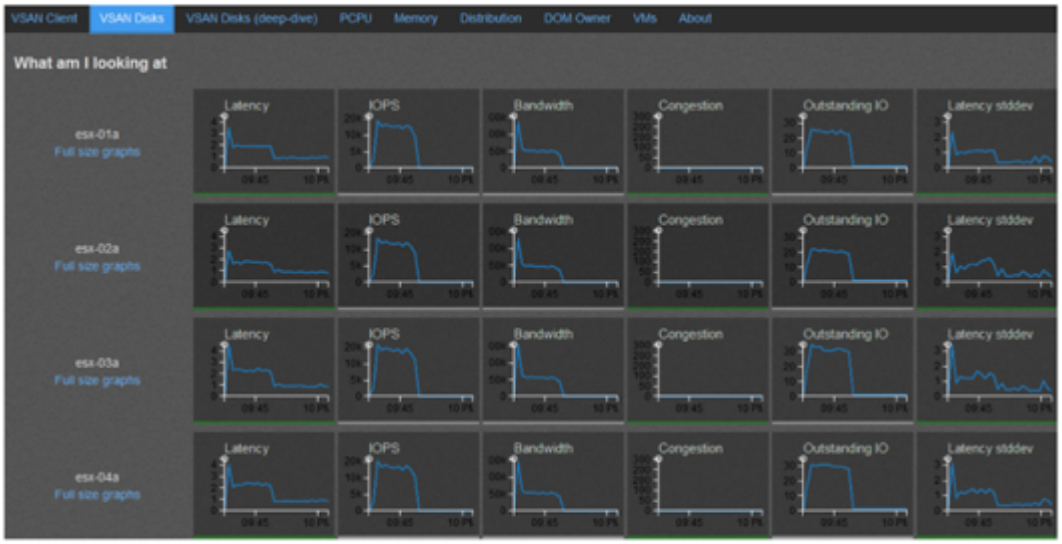


We see that

- Latencies are pretty low at around 3ms latency mark for write intensive workload
- Latency standard deviation is also low indicating a consistent latency devoid of any spikes or fluctuations
- IOPS is roughly around 8000 and holds steady throughout the duration of the workload
- Outstanding IOs are well under check at about 20
- All hosts show similar profiles, indicating a balanced cluster

VSAN Disks Tab

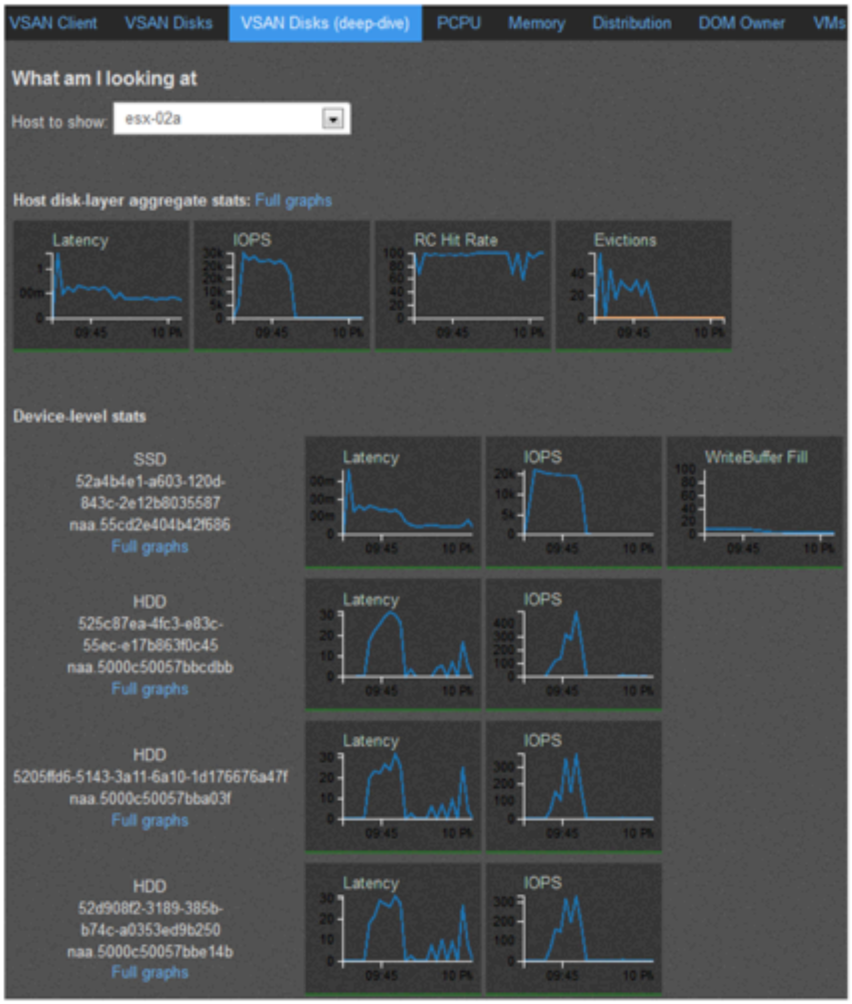
Figure 27: Virtual SAN Disks Tab



- We should recall that this layer serves IO from local disks and that the disks can be across nodes, meaning IO may to go over the network.
- If we look at the IOPS graphs we can glean that the IOPS in each host is roughly double of what we saw in the previous set of graphs in the Virtual SAN Client layer. This is due to the fact that this layer implements RAID and sync
- The corresponding average bandwidth is also roughly double of what it is in the Virtual SAN Client layer

VSAN Disks (Deep-Dive)

Figure 28: Virtual SAN Disks (Deep Dive)



Host to show: esx-02a

SSD
52a4b4e1-a603-120d-843c-2e12b8035587
naa.55cd2e404b42f686

HDD
525c87ea-4fc3-e83c-55ec-e17b863f0c45
naa.5000c50057bbcdbb

HDD
5205ffd6-5143-3a11-6a10-1d176676a47f
naa.5000c50057bba03f

HDD
52d908f2-3189-385b-b74c-a0353ed9b250
naa.5000c50057bbe14b

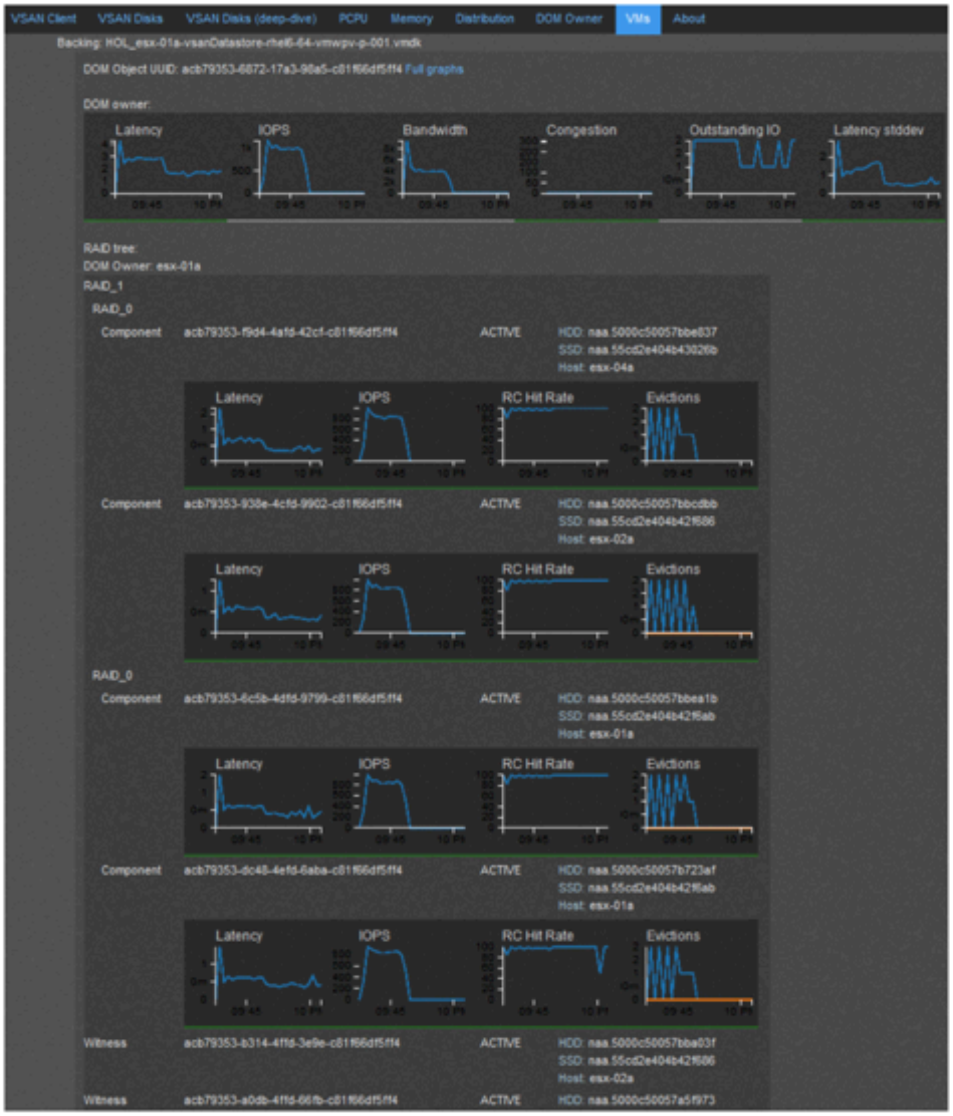
Peeking into one of the hosts, esx-02a

- We see that the overall latency in this host is hovering around 0.5ms range
- The latency of SSD is less than 1ms, however, the magnetic disks (HDD) show a latency of more than 20ms
- This shows one of Virtual SAN's features at work. The SSD in each disk group hides the high latency of magnetic disks completely
- None of the higher layers "see" the latencies of HDDs. The only detect the latency of SSD
- Write buffer is less than 10% utilized indicating fast destaging of writes to the

hard disks.

VMs Tab

Figure 29: VMs Tab



Backing: HOL_esx-01a-vsanDatastore-rhel6-64-vmwpv-p-001.vmdk

DOM Object UUID: acb79353-6872-17a3-98a5-c81f66df5ff4

DOM owner:

RAID tree:
DOM Owner: esx-01a
RAID_1
RAID_0

```
Component  acb79353-f9d4-4afd-42cf-c81f66df5ff4  ACTIVE  HDD: naa.5000c50057bbe837
                                                         SSD: naa.55cd2e404b43026b
                                                         Host: esx-04a
Component  acb79353-938e-4cfd-9902-c81f66df5ff4  ACTIVE  HDD: naa.5000c50057bbcdbb
                                                         SSD: naa.55cd2e404b42f686
                                                         Host: esx-02a
```

RAID_0

```
Component  acb79353-6c5b-4dfd-9799-c81f66df5ff4  ACTIVE  HDD: naa.5000c50057bbea1b
                                                         SSD: naa.55cd2e404b42f6ab
                                                         Host: esx-01a
Component  acb79353-dc48-4efd-6aba-c81f66df5ff4  ACTIVE  HDD: naa.5000c50057b723af
                                                         SSD: naa.55cd2e404b42f6ab
                                                         Host: esx-01a
Witness    acb79353-b314-4ffd-3e9e-c81f66df5ff4  ACTIVE  HDD: naa.5000c50057bba03f
                                                         SSD: naa.55cd2e404b42f686
                                                         Host: esx-02a
Witness    acb79353-a0db-4ffd-66fb-c81f66df5ff4  ACTIVE  HDD: naa.5000c50057a5f973
```

- Here we are looking at the VM backing disks
- We see that the VMDK object has 4 components since it has been set up with FTT of 1 and stripe width of 2
- All components are functioning fine with very low latencies
- We have some read cache misses but overall the read cache hit rate is good

From the analysis we can possibly infer

- Stripe width of 2 helps keep overall write latencies under current workload at optimal levels and helps de-stage data to disks faster
- Increasing the number of disk groups should improve latency and IOPS
- Increasing the number of disk groups will also provide more SSDs and also increase read cache size and write buffer size
- Ensuring that the number of disks in each disk group is the same is a good practice
- Also, ensuring that storage controller has a large queue depth helps keep latencies in control

Virtual SAN Observer Analysis Sample III

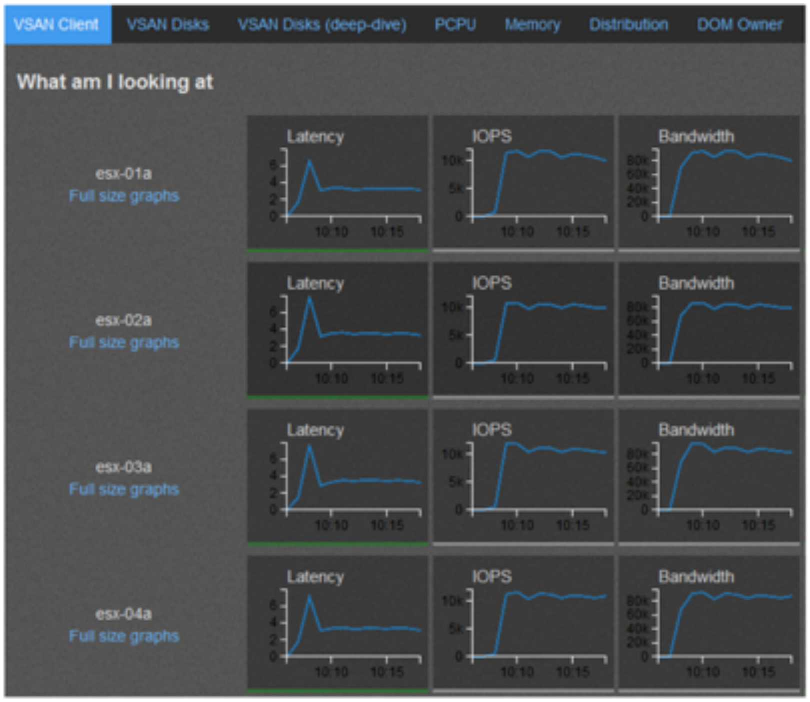
In this chapter we will look at a IO workload of 4k IO size performing 80% read and 20% write with 4 outstanding IOs

8k IO Size (80% Read, 8 Outstanding IOs)

In this example we have a workload running 8192-byte I/O that are predominantly reads (80%) with 8 outstanding IOs per VM at any point. We will take a look at some of the tabs in the UI corresponding to this workload and analyze them. Please follow this analysis by opening "Analysis_3.html" file on the desktop.

VSAN Client Tab

Figure 30: Virtual SAN Client Tab

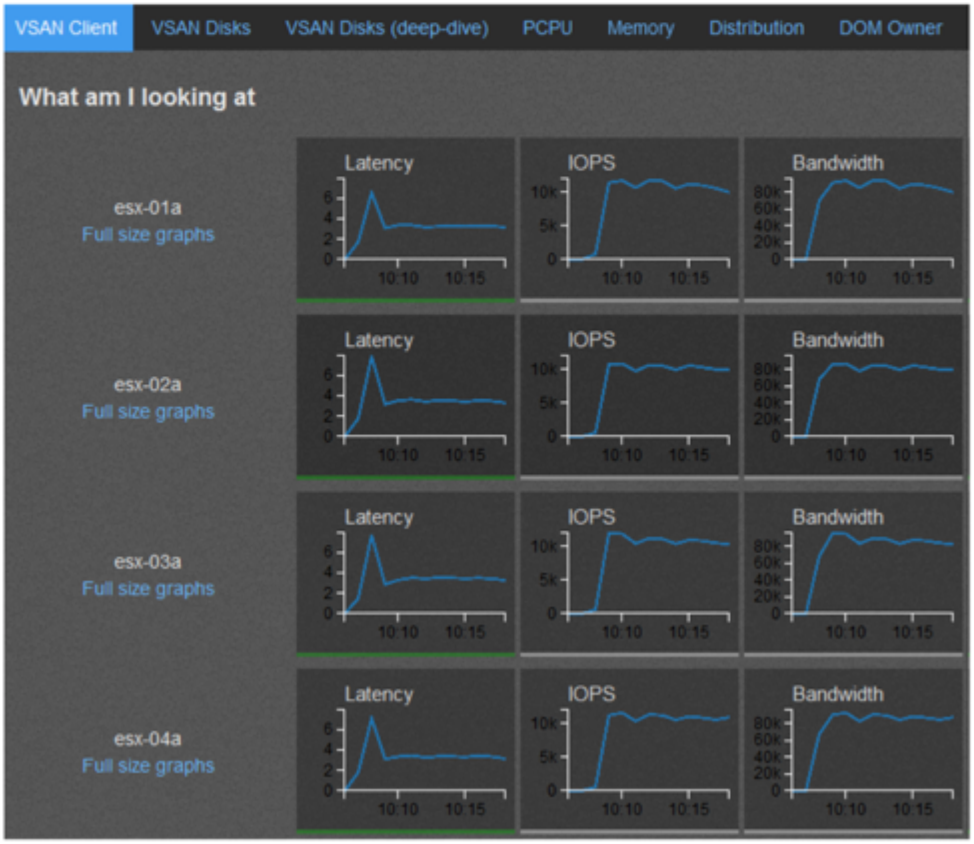


- We can observe that Virtual SAN Observer Analysis sample 3 Latency is healthy and under 4ms
- IOPS hovers around 11000 to 12000 mark
- As with the first example in Analysis 1 there is a brief latency spike

- Corresponding to some read cache misses, but soon settles down at around 3ms
- Compared to 4k 80% reads (Analysis 1) we see that bandwidth increases due to larger IOs.
- Bandwidth in this case is roughly 90MBps vs 50MBps for the analysis presented in the first analysis
- All graphs in this tab are green and performing well within threshold.
- We invite you to explore the full size graphs and delve closer into each graph by clicking on the "Full size graph" link under each host.

VSAN Disks Tab

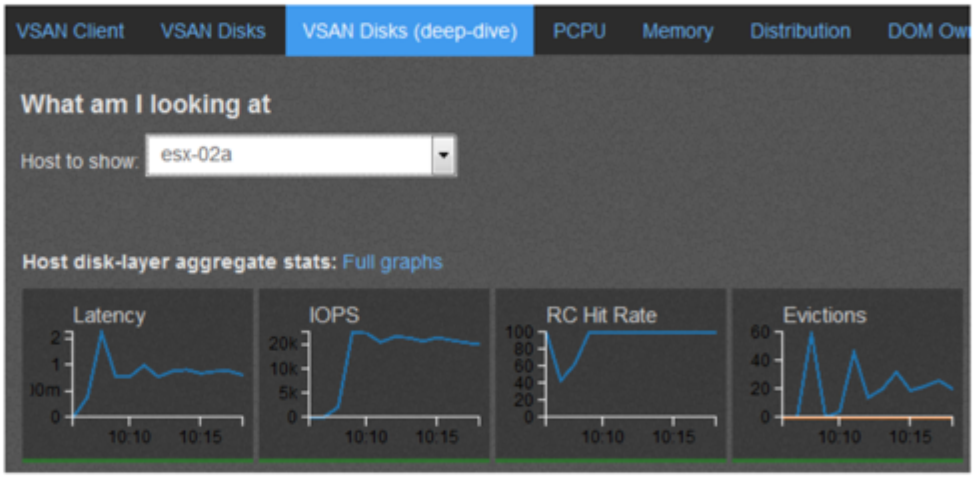
Figure 31: Virtual SAN Disks Tab



- As with the layer above all parameters are performing well at this layer
- IOPS is slightly higher than in the earlier layer as this layer includes RAID1 traffic for the 20% writes

VSAN Disks (deep-dive)

Figure 32: Virtual SAN Disks Deep Dive Tab



What am I looking at

Host to show: esx-02a

Host disk-layer aggregate stats: Full graphs

- Latency is very good. There is a small spike at the same time the random IO workload kicks in.
- When the workload starts it causes a dip in read cache hit rate due to read IOs going to uncached blocks. This causes the spike in latency explained in the previous point
- Overall read cache hit rate is around 100% for the most part, which is healthy

Virtual SAN Observer Analysis Sample IV

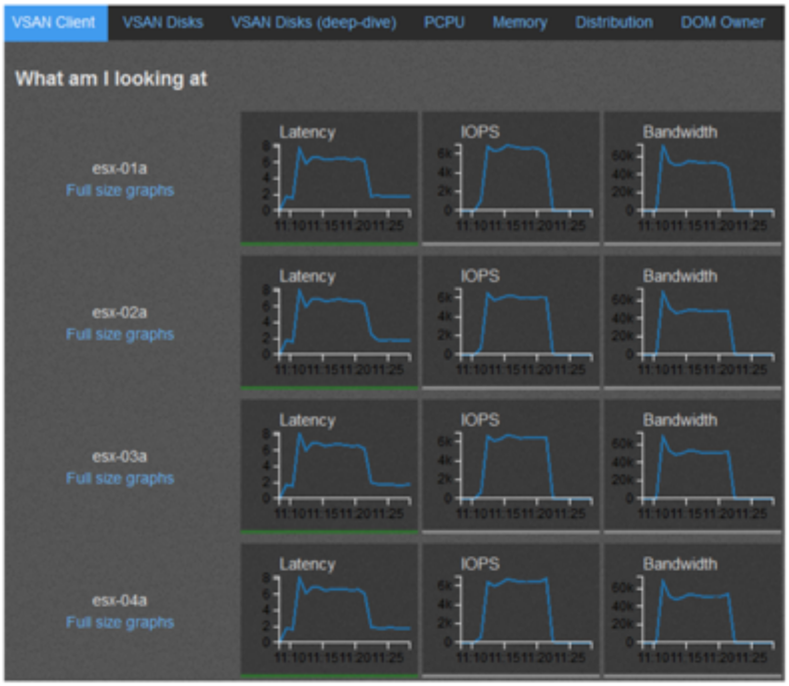
In this chapter we will look at a 8k 80% write workload with 8 outstanding IOs

8k IO Size (80% Write, 8 Outstanding IOs)

- In this example we have a workload running 8192-byte I/Os that are predominantly writes (80%) with 8 outstanding I/O per VM at any point.
- We will look at some of the important tabs in the UI corresponding to this workload.
- We invite users to delve deeper by clicking on the full graphs wherever applicable for better understanding.
- Please click on the shortcut named "Analysis_4.html" as depicted in the screen-shot
- We request you to follow our analysis on the browser by navigating to the same tab as the subsequent steps.

VSAN Client Tab

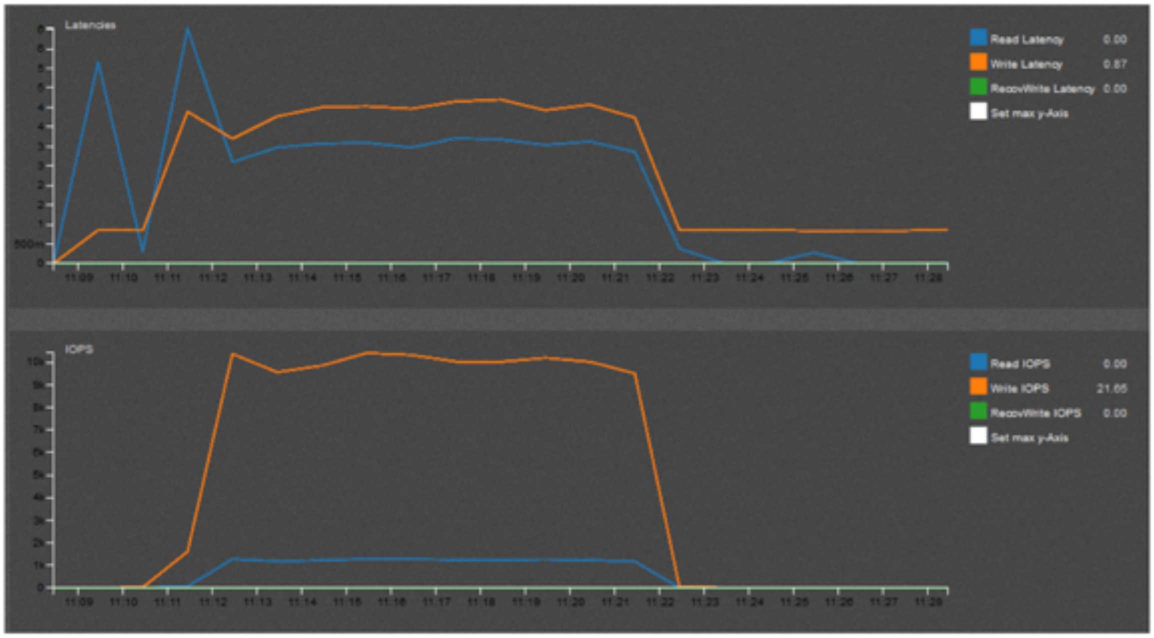
Figure 33: Virtual SAN Client Tab



- Latencies for this write workload are well under threshold values
- The hosts are doing around 7k cumulative (write + read) IOs
- The system overall looks healthy and performing well within optimal range
- We invite to user to explore other thumbnail graphs on this page and also the full size graphs under each host

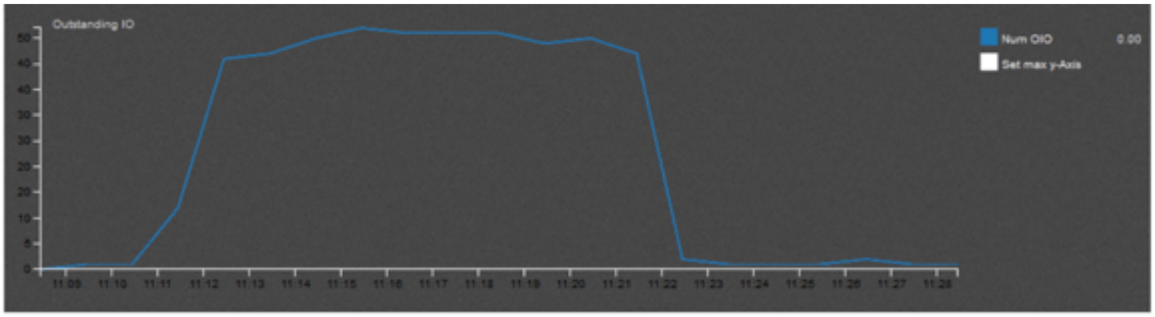
VSAN Disks

Figure 34: Virtual SAN Disks Extended Tab



- Here we show full size graphs from host esx-01a under VSAN Disks tab
- We see that both read and write latencies are around 4ms for most part
- Initial read latency spike is, as explained earlier, due to reading uncached blocks from disk
- The system is doing about 11k IOPS overall

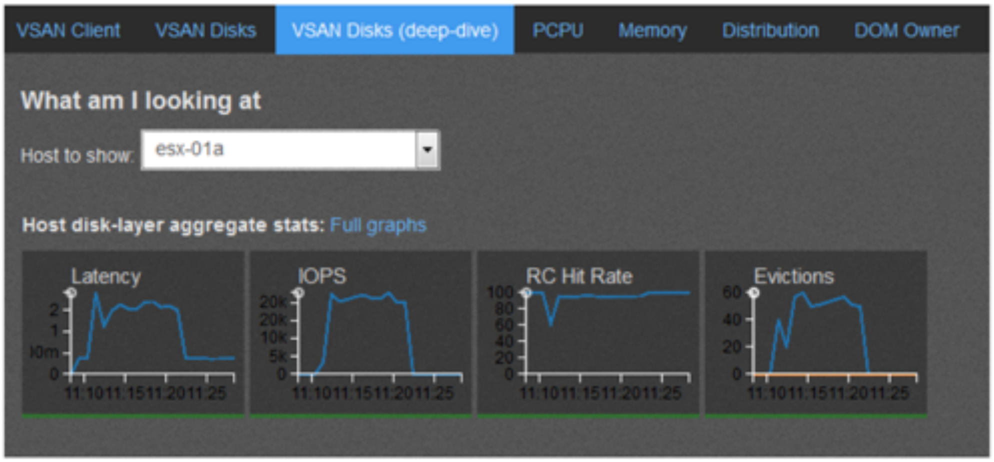
Figure 35: Virtual SAN Disks Extended View



- We continue looking at full size graphs of host esx-01a under VSAN Disks tab
- Here we are looking at outstanding IO
- Each VM is issuing 8 outstanding IOs and there are 8 VMs in each host
- Cumulatively this layer is handling about 50 outstanding IOs on this host
- The storage controller (LSI 2308) has a large queue depth (default of 600), which helps keep latencies low

VSAN Disks (deep-dive) tab

Figure 36: Virtual SAN Disks (Deep Dive)



What am I looking at

Host to show: esx-01a

Host disk-layer aggregate stats: Full graphs

- In our final step we take a look at VSAN Disks (deep-dive) tab
- As with previous layers we see that all parameters are performing well in this
- write heavy workload
- Read cache hit rate dips, as expected, around the time random workload starts by accessing uncached blocks. But immediately the read cache hit rate reached 100%
- In spite of some evictions the read cache hit rate is healthy
- We conclude our analysis of this workload here, as we have determined that
- the system is performing optimally

Acknowledgments

Would like to thank Christian Dickmann, Staff Engineer of the VMware R&D Team, and also Mousumi Millick, Manager of Technical Staff in VMware QE Team, for their contributions to this paper. Joe Cook, Sr. Technical Marketing ManagerCharu Chaubal, Group Manager of the Storage and Availability Technical Marketing team for reviewing this paper and making it look sexy.

About the Author

Rawlinson Rivera is a Senior Architect in the Cloud Infrastructure Technical Marketing Group at VMware focused on Software-Defined Storage technologies primarily responsible for Virtual SAN, Virtual Volumes, and OpenStack.

As a previous Architect in VMware's Cloud Infrastructure & Management Professional Services Organization, Rawlinson specialized on vSphere and Cloud enterprise architectures for VMware's fortune 100, 500 customers.

Rawlinson is amongst the few VMware Certified Design Experts (VCDX#86) in the world, and author of multiple books based on VMware and other technologies.

Follow Rawlinson’s blogs:

- http://blogs.vmware.com/vsphere/storage
- http://www.punchingclouds.com

Follow Rawlinson on Twitter:

- @PunchingClouds