IntelliFlash™ Best Known Methods

VMware vSphere®

Table of Contents

Overview

Concerns about virtual server performance often prevent certain enterprise applications from being virtualized. Performance also determines the number of VMs (VMs) that can be consolidated within a physical server, impacting the overall cost of virtualization.

Tintri IntelliFlash systems deliver low latency and high IOPS for virtual server environments running performance-hungry applications such as VDI, SQL Server®, Share Point®, Oracle®, and other business-critical applications. IntelliFlash systems deliver 5x the performance of traditional storage solutions, allowing you to virtualize more enterprise applications and achieve better VM consolidation.

IntelliFlash systems support multiple protocols simultaneously – FC, iSCSI, NFS, and SMB. A user can choose to provision storage for VMs using block storage or file protocols such as NFS and SMB3. In addition to VMs, you can also provision file shares for the VMs using NFS or from a single IntelliFlash system.

With the IntelliFlash's patented metadata acceleration transaction wait times are dramatically reduced by orders of magnitude and real-time deduplication and compression act as performance multipliers by maximizing the use of flash storage. This can reduce your storage capacity needs for virtual servers by at least 75% which cut down your storage acquisition and operational costs. And all the additional features are available, from replication to data management capabilities such as snapshots and clones.

IntelliFlash systems are certified with all the major hypervisors, including VMware ESXi™, Microsoft Hyper-V, and Citrix® XenServer®. Joint development and integration with hypervisors through various APIs enables you to deploy your virtual server infrastructure with confidence and peace of mind. Integration also makes your virtual infrastructure management seamless, and allows you to snapshot, backup, and restore your VMs and data in a VM-consistent and application-consistent manner.

Scope

The Tintri by DDN IntelliFlash Best Known Methods for VMware vSphere document describes the best known methods for using IntelliFlash systems in vSphere 5.5+ and 6+ environments. IntelliFlash systems cover general support for a wide variety of VMware products; this guide will focus on best known methods related to VMware ESXi & VMware vCenter®.

The document covers the best known methods for the following:

- Pre-deployment planning
- Network connectivity
- Storage provisioning
- Host networking and SAN/NAS configuration
- VM provisioning, configuration, and maintenance operations
- Snapshot and replication

In addition, it also provides best known methods in the following areas:

- Performance tuning
- Troubleshooting

Target Audience

This document is intended to provide understanding and insight of the best known methods when using VMware vSphere with Tintri IntelliFlash storage systems.

This guide is intended for system, storage, and database administrators, and implementation engineers who manage, monitor, or deploy virtual servers and storage solutions in the data center.

This document assumes that the audiences have a working familiarity with VMware® concepts and products such as ESXi, vSphere, vCenter, VMs, and related operations. Also a familiarity with networking and storage systems is beneficial.

Planning Ahead

There is a lot of conventional wisdom around virtualization best practices. Here are some of the categories in the context of deploying IntelliFlash you'll want to consider before implementing your virtualized environment.

VM and Applications

Review the following best methods to balance the VMs and applications on the host and storage:

- Consider the compute and memory usage of the VMs. Balance the VMs on multiple hosts based on compute and memory usage.
- Consider the I/O load of the VM. Balance the VMs on IntelliFlash systems across controllers, Pools, and LUNs/shares based on I/O load. For example:
    - o Domain controllers usually have a light I/O load; multiples of this type of VMs can be allocated on a single LUN/share.
    - o Large SQL Server databases usually have heavy I/O load. It is recommended to configure dedicated LUNs/shares for I/O-intensive applications such as large SQL Server databases.
- Group the VMs based on backup policies. Consider provisioning VMs with the same backup policy on the same set of IntelliFlash LUNs/shares.

Networking

For optimal resilience, performance and manageability, it's recommended to use separate IP networks and network ports for management, vMotion, SAN/NAS storage network and remote replication. This is also consistent with VMware recommendations.

IntelliFlash Management Network

Consider the following management network related best methods:

- To configure the management network on an IntelliFlash system, you need five IP addresses, along with information about the netmask, gateway, and DNS. Please obtain this information from your network administrator prior to deploying the system.
- Out of the five IP addresses, two will be used for IPKVM, two for local management ports on each controller, and one for the floating High Availability (HA) IP address used by management services, such as IntelliFlash VMware Client Plugin (VCP) for vCenter.
- IntelliFlash supports network path redundancy for management, which provides management service availability in the case of network path failure. If you desire such levels of availability, consider connecting two of the onboard physical ports per controller for management. On older IntelliFlash models (HA2000 and T3000 models) the onboard ports were 1GbE. On newer IntelliFlash models (T4000 and N-5000) the ports are 10GbE. Some customers enjoy having an extra 10GbE port to use for various Projects so only a single port is used for management.

Storage Network for iSCSI and NFS

Consider the following iSCSI and NFS network related best methods:

- Use separate subnets for storage data and management networks.
- On each ESXi host, configure a minimum of two data network ports for redundancy purposes.
- If both iSCSI and NFS are used in the same environment, it's recommended to use two separate subnets: one subnet for iSCSI, the other for NFS.
- If iSCSI and NFS share the same physical NICs and network paths on the host or on the IntelliFlash system, consider configuring VLANs to segment the network to allow different subnets for iSCSI and NFS.
- Obtain the IP addresses and netmask information ahead of time from the network administrator.
    - o For each interface group, it's recommended that you assign the same number of floating IP addresses as the number of physical ports in the group. For example, if there is one interface, we recommend at least two 10GbE ports for HA which in turn means 2 floating IP addresses.

Replication Network

- To configure the replication network on an IntelliFlash system, you will need 1 IP address per Pool/controller along with the information about the netmask, gateway, and DNS for the replication subnet.
- Use a separate interface group and subnet for replication (By default it uses the management port)
- If using the same physical network ports for different purposes, configure VLANs to segment the networks
- Consider connecting a minimum of two network ports for path redundancy. In the case of network path failure, replication can continue on the surviving network path.

Boot from SAN

IntelliFlash systems support boot from SAN by using both FC and iSCSI. Most FC HBAs support boot from SAN. For iSCSI boot from SAN support, it requires the host network interface card to support iBFT (iSCSI Boot Firmware Table). If boot from SAN is planned, make sure to select a NIC model that offers iBFT support and is certified by VMware for your particular version of ESXi.

IntelliFlash System Network Design

Now that you've considered the pre-deployment options this sections look at implementing the connections with the IntelliFlash storage system.

Ethernet Basics

Before any discussion can be had around Ethernet configuration a look at the Interface Group must come first.

Interface Group

An Interface Group consists of one or more physical ports with at least one IP address assigned to it. This offers path redundancy and load balancing across the physical interface members of the group. It also improves overall network performance if there are multiple IP data streams with different destination IP addresses. In such situations, interface groups spread outbound network traffic across the interfaces in the interface group. Interface group configuration is available through the IntelliFlash Web UI, under Settings > Network > Interface.

Floating IP Address

Floating IP addresses work together with the interface group to redundancy across the storage controllers in the case of controller failover and failback.

Floating IP addresses can be configured through the IntelliFlash Web GUI under Settings > HA or through the Network page. A floating IP address belongs to an HA resource group, and is owned by one controller at any given time.

By default one HA floating IP address is created during installation with the Initial Configuration Wizard (ICW). This floating IP address is used for management (See Management Networks) by the IntelliFlash VMware Client Plug-in (VCP) and the Tegile Data Protection Service (TDPS) to communicate with the system.

Management Networks

By default, the IntelliFlash system has one pre-configured interface group for management called mgmt0. The interface group mgmt0 initially consists of two physical network interfaces. You can modify the management interface group and assign it to any port available on the system however IPKVM access is only available on the embedded port0 – igb0 or ixgbe0.

There are 3 IP addresses associated with this group – the high availability (HA) floating (See Floating IP Address) address, which is the dedicated management address and will move between controllers during maintenance or failovers. The other 2 IP addresses are fixed and assigned to each controller.

Figure 1 shows a high level illustration of a simple network topology for configuring an IntelliFlash system on the management network. In some environments, if there are multiple network switches for your management network, it's recommended to connect the switches with inter-switch links.

On the network switches, enable "Port Fast" or the equivalent to disable spanning tree on the ports that are connected to the system management ports.

For path redundancy purposes, connect two ports per controller to the management network.

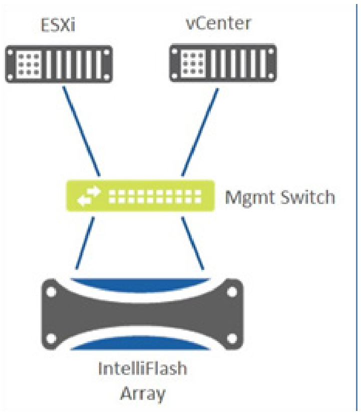


Figure 1 - IntelliFlash Management Network

VLAN

If you are sharing the same physical network ports for both management and replication, Tintri recommends segmenting the network by configuring two VLANs: one for management, and the other for replication.

You can add or edit the VLAN's interfaces under the interface group mgmt0 through the IntelliFlash Web UI, under Settings > Network > Interfaces.

iSCSI and NFS Networks

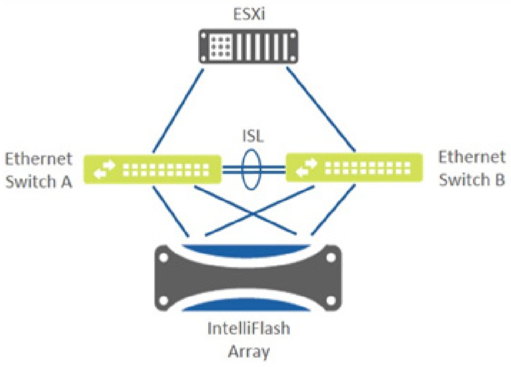


Figure 2 - IntelliFlash iSCSI / NFS Stacked Network

Figure 2 illustrates a fully redundant topology between the host and both controllers of the IntelliFlash system, with redundant paths from host to storage through redundant storage network switches. This is an example of stacked switch topology.

Stacked switch topology can be used on both NFS and iSCSI. In such a scenario, Tintri recommends configuring the IntelliFlash network configuration for NFS or ISCSI as follows:

- A minimum of two physical ports per controller is recommended: one port connected to switch A and the other port connected to switch B
- Create minimum of one Interface group
- Assign minimum two ports, Physical or VLAN, per Interface group
- Configure minimum of two floating IP address per Interface group. The number of floating IP address should be the same as the number of ports in the Interface group. Usually for Stacked switch configuration, these IP addresses are on the same subnet.

Non-stacked switch topology can be configured for iSCSI, but not recommended for NFS, in VMware ESXi environments. It's recommended to configure the IntelliFlash network configuration for iSCSI as follows:

- Create minimum of two Interface groups
- For systems running IntelliFlash OS 3.5 or a later version,
  - o A minimum of two ports per controller is recommended: one port connected to switch A, and the other port connected to switch B, similar to what Figure 2 illustrated.
  - o These two ports should be placed in the same Interface group with failover policy set to "Wait until all IPs fail".
- Configure a minimum of two floating IP addresses per Interface group. The number of floating IP address should be the same as the number of ports in the Interface group.
  - o Each Interface group must be on its own separate subnet
  - o The IP addresses associated with an Interface group must be in the same subnet

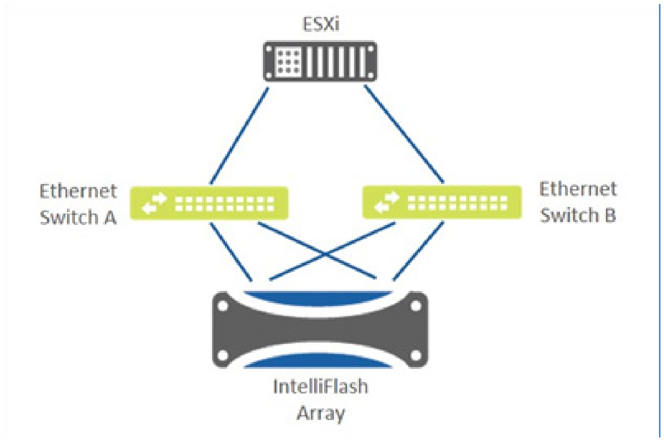


Figure 3 - IntelliFlash iSCSI / NFS Non-Stacked Network

Jumbo Frames

Enable jumbo frames end-to-end including the switch ports that are connected to the ESXi hosts and the IntelliFlash system.

Note: You should ensure that jumbo frame settings are persistent across switch reboots. On configuration of jumbo frames end-to-end, please check the jumbo frame functionality by using the instructions in the Troubleshooting section of this document under Testing Jumbo Frame.

Flow Control

In general, it's recommended to enable bi-directional flow control end-to-end on the SAN/NAS network. You should enable flow control on the networks on the IntelliFlash system, on the ESXi host, and on the related switch ports.

Note: You should ensure that the flow control settings are persistent across switch reboots.

Other Recommendations

In addition to the above, apply the following best practices:

- Enable "Port Fast" or the equivalent to disable spanning tree on the switch ports that are connected to the hosts and the IntelliFlash system.
- Make sure these switch settings are persistent across switch reboots.
- For additional recommendations about iSCSI/NFS deployments on Cisco UCS, please refer to the IntelliFlash UCS Configuration and Best Practices Guide.

Fibre Channel Network

Figure 4 illustrates a fully redundant topology between the host and both controllers of the IntelliFlash system, with redundant paths from host to storage through redundant FC switches.

FC Switch Zoning

A one-to-one zoning configuration between each FC port on each host and each FC port of each controller on the IntelliFlash system is conventional wisdom but it is not necessary as Fibre Channel is quite mature at this point. Zoning is typically done once so it only adds a onetime level of complexity and it can help in troubleshooting.

To achieve one-to-one zoning as in the physical topology illustrated in Figure 4, the following WWN based zones should be configured:

- Host FC port 0 to storage controller A FC port 0
- Host FC port 0 to storage controller B FC port 0
- Host FC port 1 to storage controller A FC port 1
- Host FC port 1 to storage controller B FC port 1

Repeat the above zoning configuration for each host that requires access to the IntelliFlash system through FC. If multiple systems are used, repeat the above zoning configuration for each IntelliFlash system.

For additional recommendations about FC deployment on Cisco UCS, please also refer to the IntelliFlash UCS Configuration and Best Practices Guide.

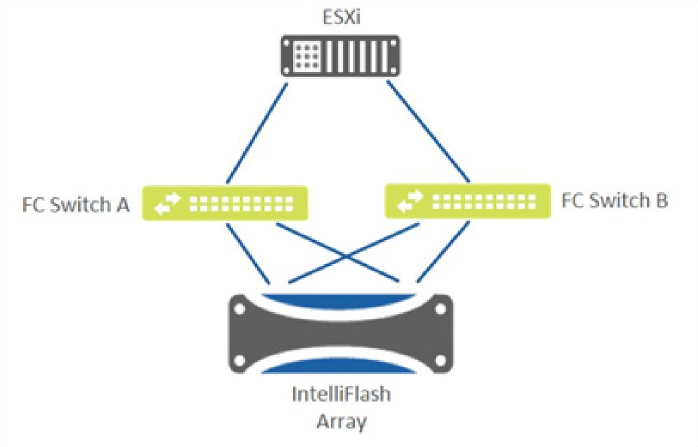


Figure 4 – IntelliFlash FC Network

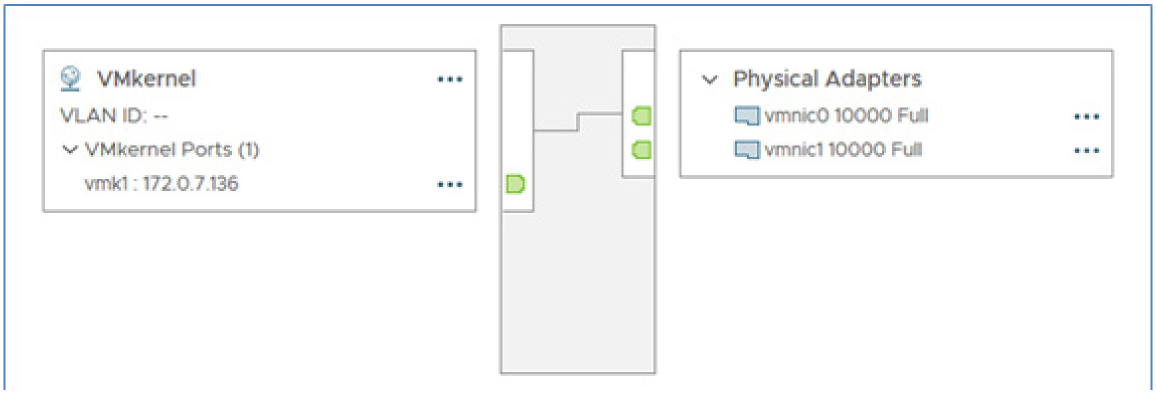
ESXi and VMware vCenter Configuration

This section covers all aspects of configuring the ESXi host's network to connect to the IntelliFlash storage system. It covers all the protocol networks and in addition highlights how the IntelliFlash VMware Client Plugin (VCP) can also be used to configure the ESXi host.

Network Configuration

NFS Network Configuration

NFS is deployed on a single subnet in most deployments. You should configure a vSwitch with a minimum of two physical network adapters for path redundancy, as shown in the following screenshot:



VMkernel
VLAN ID: --
VMkernel Ports (1)
vmk1 : 172.0.7.136

Physical Adapters
vmnic0 10000 Full
vmnic1 10000 Full

Figure 5 - vSwitch With 2 Physical Ports

In the vSwitch, create one VMKernel port group that uses all the physical network adapters on the vSwitch.

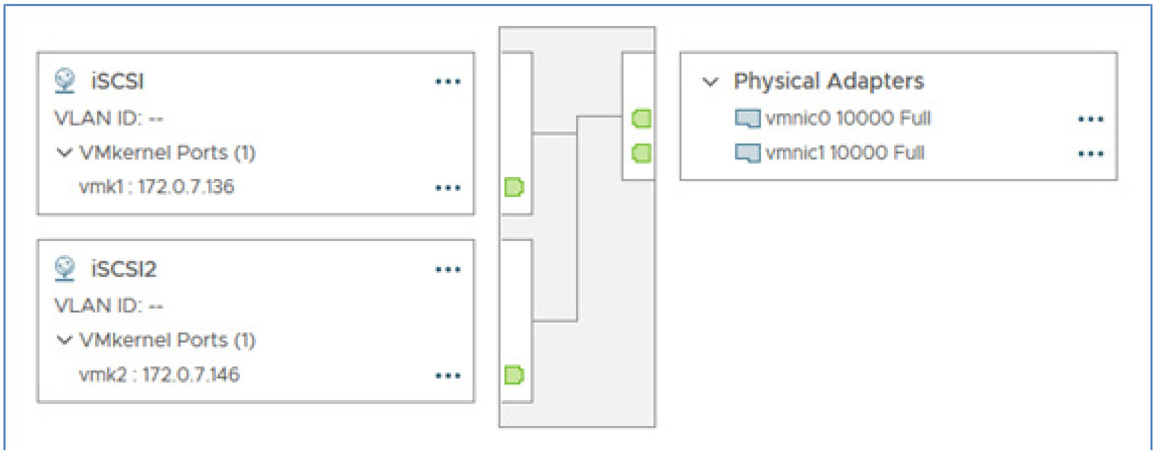
iSCSI Network Configuration

iSCSI On One Subnet

When all iSCSI network adapters are on a single subnet, you should configure one vSwitch using a minimum of two physical network adapters for path redundancy, and create a minimum of two VMKernel port groups (Figure 6).

This setup requires that only one physical port be active in each VMKernel. Having a port in standby mode causes the ports to behave like network ports and not storages ports. Here's an informative VMware article discussing this topic: Why can you not use NIC Teaming with iSCSI Binding?

Even though this is not NIC Teaming the configuration is achieved through NIC Teaming UI or CLI commands. Please follow the steps below:



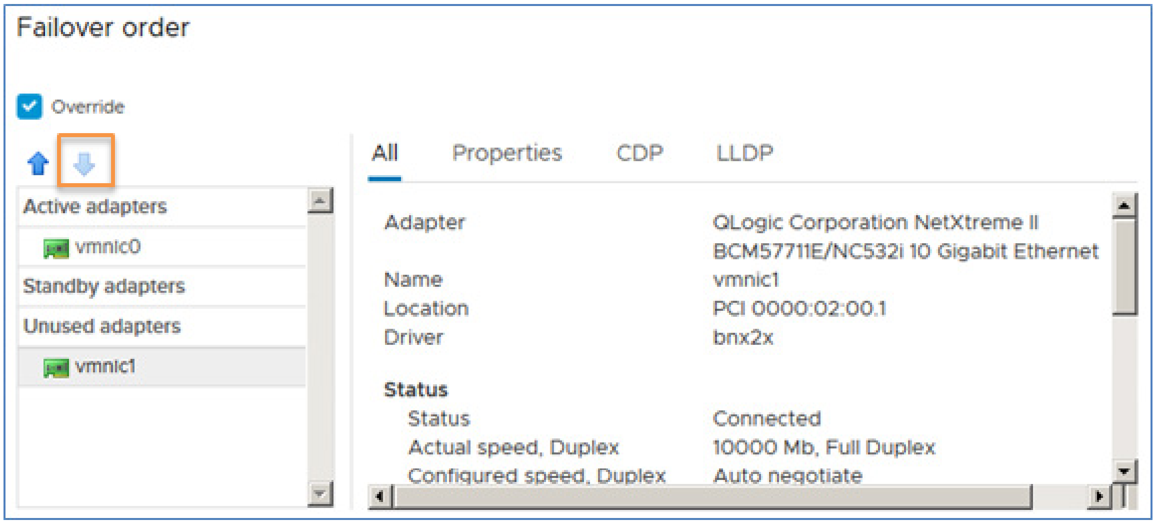
iSCSI
VLAN ID: --
VMkernel Ports (1)
vmk1 : 172.0.7.136

iSCSI2
VLAN ID: --
VMkernel Ports (1)
vmk2 : 172.0.7.146

Physical Adapters
vmnic0 10000 Full
vmnic1 10000 Full

Figure 6 - Storage Failover through NIC Teaming

In vSphere 6.7, this can be achieved through the UI or CLI. The UI is very intuitive as shown in Figure 7. Simply edit the port group and in the Teaming and failover section check Override and then use the arrows to move the vmnic into the Unused Adapters category.



Failover order

☑ Override

Active adapters
- vmnic0

Standby adapters

Unused adapters
- vmnic1

Adapter	QLogic Corporation NetXtreme II BCM57711E/NC532i 10 Gigabit Ethernet
Name	vmnic1
Location	PCI 0000:02:00.1
Driver	bnx2x
Status	
Status	Connected
Actual speed, Duplex	10000 Mb, Full Duplex
Configured speed, Duplex	Auto negotiate

Figure 7 - Configuring Active Adapters through NIC Teaming

To do this using the CLI follow the steps below:

```
esxcli network vswitch standard portgroup policy failover set –p iSCSI –a vmnic0
```

By specifying explicitly only one vmnic as active ESXi bumps the other vmnic into the Unused Adapter category. Run the following command to verify the settings:

```
esxcli network vswitch standard portgroup policy failover get –p iSCSI

Load Balancing: srcport
Network Failure Detection: link
Notify Switches: true
Failback: true
Active Adapters: vmnic0
Standby Adapters:
Unused Adapters: vmnic1
Override VSwitch Load Balancing: false
Override VSwitch Network Failure Detection: false
Override VSwitch Notify Switches: false
Override VSwitch Failback: false
Override VSwitch Uplinks: true
```

Repeat the same procedure on the second VMkernel port group. In this example, on VMKernel port group iSCSI2, we need to set vmnic1 into Active Adapters and vmnic0 into Unused adapters by issuing the following esxcli command:

```
esxcli network vswitch standard portgroup policy failover set –p iSCSI2 –a vmnic1
```

Note: There is a bug in vSphere 6.5a GA with NIC teaming where the option to add an adapter as Unused doesn't exist. There are only two options, Active Adapters and Standby Adapters. (Refer https://labs.vmware.com/flings/esxi-embedded-host-client/bugs/298)

iSCSI On Multiple Subnets

If iSCSI network adapters are on multiple subnets, a separate vSwitch should be configured for each subnet. Please note that iSCSI initiator port binding must not be configured if you have iSCSI network adapters on multiple subnets.

Please see more recommendations about how to configure networking in Best Practices for Running VMware vSphere on iSCSI.

VLAN

If a specific VLAN is designated for NFS or iSCSI, configure the VMKernel port groups for them with the appropriate NFS or iSCSI VLAN tag(s).

Jumbo Frame

Tintri recommends enabling jumbo frame by setting the MTU to 9000 on all the vSwitch and VMKernel port groups that the ESXi host uses to access the IntelliFlash system. This applies for both for iSCSI and for NFS.

After configuring jumbo frame on each of the individual components, make sure to test if jumbo frame is configured end-to-end. See jumbo frame test tips in the Troubleshooting section.

Flow Control

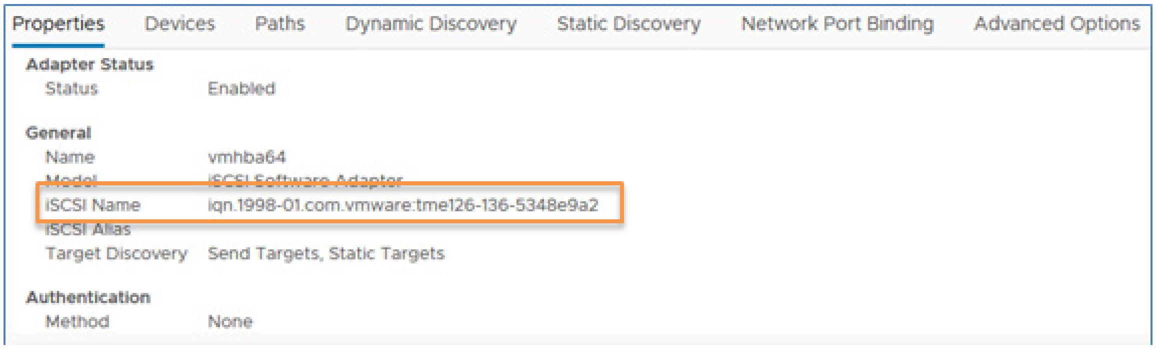
It's recommended to enable flow control on all the network adapters that ESXi host uses to access the IntelliFlash system, both for iSCSI and for NFS.

Please note that even though flow control should be enabled by default on ESXi, there are still chances that flow control can be off due to auto-negotiation. Therefore, it's recommended to check the flow control status on the network adapters for iSCSI and NFS, and enable it as necessary.

Software iSCSI Initiator Configuration

iSCSI Initiator Name

Add the software iSCSI adapter on the ESXi host and make a note of the iSCSI initiator name for later use in configuring the initiator on IntelliFlash system SAN configuration.



Adapter Status

Status	Enabled

General

Name	vmhba64
Model	iSCSI Software Adapter
iSCSI Name	iqn.1998-01.com.vmware:tme126-136-5348e9a2
iSCSI Alias	
Target Discovery	Send Targets, Static Targets

Authentication

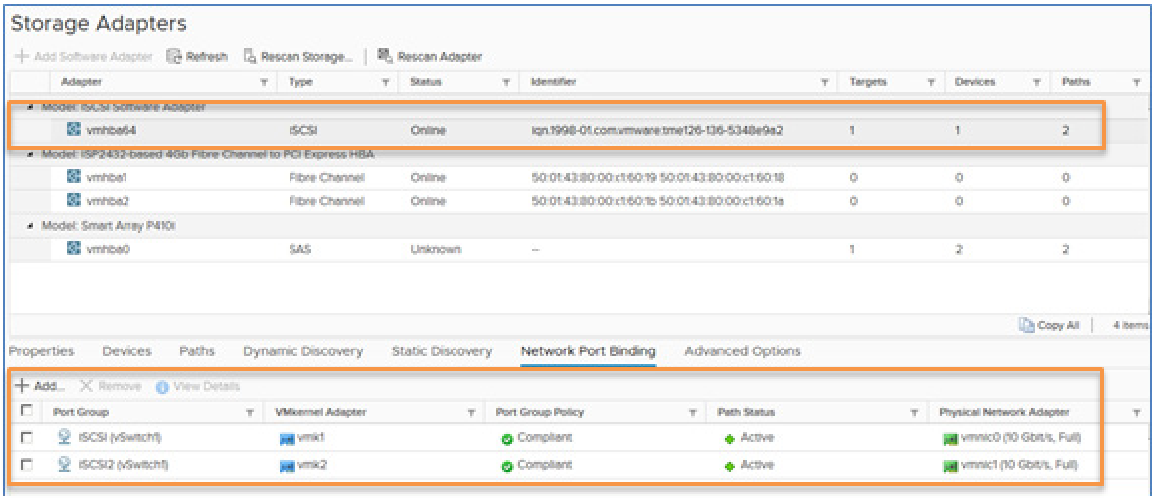
Method	None

Figure 8 - iSCSI Name

iSCSI Initiator Port Binding

iSCSI Initiator Port Binding is only applicable to the configuration when all iSCSI network ports are on one subnet. If you have your iSCSI network adapters on multiple subnets, please skip this sub¬section.

In the vSphere web client, under Storage > Storage Adapters, click on the iSCSI software adapter and then select the Network Port Binding tab. Bind the iSCSI VMkernel port groups to the software iSCSI initiator by clicking on the Add button.



Adapter	Type	Status	Identifier	Targets	Devices	Paths
Model: iSCSI Software Adapter						
vmhba64	iSCSI	Online	iqn.1998-01.com.vmware:tme126-136-5348e9a2	1	1	2
Model: ISP2432-based 4Gb Fibre Channel to PCI Express HBA						
vmhba1	Fibre Channel	Online	50:01:43:80:00:c1:60:19 50:01:43:80:00:c1:60:18	0	0	0
vmhba2	Fibre Channel	Online	50:01:43:80:00:c1:60:1b 50:01:43:80:00:c1:60:1a	0	0	0
Model: Smart Array P410i						
vmhba0	SAS	Unknown	--	1	2	2

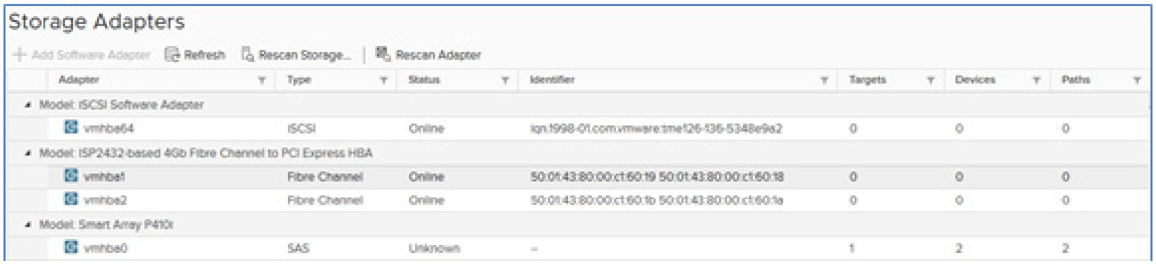
Port Group	VMkernel Adapter	Port Group Policy	Path Status	Physical Network Adapter
iSCSI (vSwitch1)	vmk1	Compliant	Active	vmnic0 (10 Gbit/s, Full)
iSCSI2 (vSwitch1)	vmk2	Compliant	Active	vmnic1 (10 Gbit/s, Full)

Figure 9 - Configuring iSCSI Initiator

ESXi FC Adapter WWPNs

Make a note of the WWPNs of the FC adapters on each of the ESXi hosts. You can find the WWPNs using the vCenter web client. Select the ESXi server and click on Configure > Storage > Storage Adapters.

You will need the WWPNs when you configure the FC SAN portion on the IntelliFlash system.



Adapter	Type	Status	Identifier	Targets	Devices	Paths
Model: iSCSI Software Adapter						
vmhba64	iSCSI	Online	iqn.1998-01.com.vmware:tme126-136-5348e9a2	0	0	0
Model: ISP2432-based 4Gb Fibre Channel to PCI Express HBA						
vmhba1	Fibre Channel	Online	50:01:43:80:00:c1:60:19 50:01:43:80:00:c1:60:18	0	0	0
vmhba2	Fibre Channel	Online	50:01:43:80:00:c1:60:1b 50:01:43:80:00:c1:60:1a	0	0	0
Model: Smart Array P410i						
vmhba0	SAS	Unknown	--	1	2	2

Figure 10 - FC World Wide Port Numbers (WWPNs)

Note: Unlike the prior versions of ESX server, the WWPN information in ESXi 6.5 and higher can only be retrieved from the vCenter web client or through ESXi server CLI. This information is no longer available in the ESXi vSphere web client. A sample CLI command is below.

```
esxcli storage san fc list
```

ESXi Host NFS Configuration

It's recommended that you validate the NFS related configuration on each of the ESXi hosts. Table 1 provides recommended values which should be set using the IntelliFlash VCP. Please see the section IntelliFlash VMware Client Plugin (VCP) for more information on how to use it.

	ESXi 5.5+Default	ESXi 6.5+Default	Suggested Value	Reboot required?
Net.TcpipHeapSize Heap memory (MB) allocated for managing VMkernel TCP/IP connectivity.	0	0	32	Yes
Net.TcpipHeapMax The max heap memory (MB) allocated for managing VMkernel TCP/IP connectivity.	256	512	ESXi 5.5: 512 ESXi 6.5: 1536	Yes
NFS.MaxVolumes[2] Limits the number of NFS datastores which can be mounted by the vSphere ESXi/ESX host concurrently.	8	8	256	Yes
NFS.MaxQueueDepth[3] Maximum per volume queue depth.	4294967295	4294967295	64	Yes
NFS.DeleteRPCTimeout[4] NFS delete timeout in seconds.	10	12	30	Yes
NFS.HeartbeatFrequency Time in seconds between heartbeats.	12	12	20	Yes

Table 1 - ESXi NFS Recommended Values[1]

Notes:

1. All values in this table are configurable through the IntelliFlash VCP.
2. As NFS.MaxVolumes is increased, the Net.TcpHeapSize and Net.TcpipHeapMax should be increased from the default accordingly.
3. For more considerations about the optimal value of NFS.MaxQueueDepth in a particular environment, please refer to the section of Understanding Queue Depth End-to-End.
4. Can no longer be set from vSphere Web Client only through CLI or IntelliFlash VCP.

You can also configure the above ESXi host settings using the VMware vSphere client through Host > Configure > System > Advanced System settings. You can also configure them through the ESXi shell or vSphere PowerCLI.

For more information about these NFS related configuration, please refer to these articles at the VMware knowledge base:

1. Definition of the advanced NFS options (1007909)
2. Increasing the default value that defines the maximum number of NFS mounts(2239)

IntelliFlash SAN/NAS Configuration

Configuring Network Settings for iSCSI and NFS

Jumbo Frames

Tintri recommends that you enable jumbo frames on physical network interfaces for iSCSI or NFS on both controllers of the IntelliFlash system.

To enable jumbo frames, the physical network interface MTU should be set to 9000. You can enable it using the IntelliFlash Web UI, under Settings > Network > Interface. When you finish configuring jumbo frame, from server, network, to storage, make sure to test jumbo frame is configured properly end-to-end. For more information, see the jumbo frame test tips in the Troubleshooting section.

Flow Control

In general, it's recommended that you enable flow control on all physical network interfaces for iSCSI or NFS on both controllers of the IntelliFlash system.

VLAN

When you share a physical network interface for multiple VLANs, for example two VLANs: one for NFS and the other for iSCSI, you can segment the physical port into multiple VLAN ports by adding VLAN links under the designated physical network interface. You can add VLAN links through the IntelliFlash Web UI under Settings > Network > Interface.

Interface Group

It's recommended that you add an Interface group for each storage subnet, and bind the designated physical network interfaces, or VLAN links to it. You should designate a minimum of two network ports per controller for your data network. This ensures the host can continue to access the storage devices on the system in the case of a network path failure.

It is possible to designate only one network interface per Interface group but it will not have any redundancy on that controller.

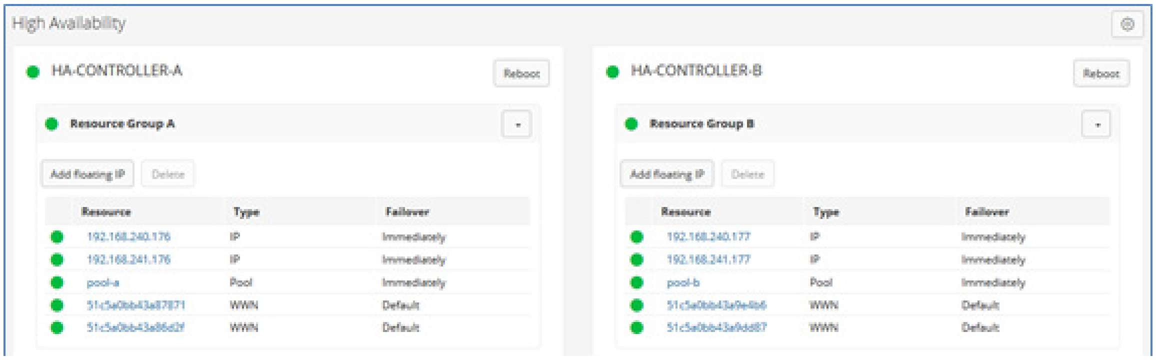
Floating IP Address

Floating IP addresses are used by the ESXi hosts to mount NFS shares or to access iSCSI LUNs. This ensures the host can continue to access the storage devices on the system in the case of controller failover.

For data networks, when using a single Pool configuration (Controller A only), you should configure the floating IP addresses on controller A only; in the case of 2 Pool configuration (Controllers A/B), an equal number of floating IP addresses should be configured on both the controllers.

Floating IP address can be configured through the IntelliFlash Web GUI under Settings > Network > Interface > Interface Groups or under Settings > High Availability. For each Interface group, it's recommended that you add the same number of floating IP addresses as the number of physical or VLAN interfaces within the Interface group. This helps balance the storage I/O load across the available network paths.

Figure 11 shows and example of an Active/Active configuration:



	Resource	Type	Failover
●	192.168.240.176	IP	Immediately
●	192.168.241.176	IP	Immediately
●	pool-a	Pool	Immediately
●	51c5a0bb43a87871	WWN	Default
●	51c5a0bb43a86d2f	WWN	Default

	Resource	Type	Failover
●	192.168.240.177	IP	Immediately
●	192.168.241.177	IP	Immediately
●	pool-b	Pool	Immediately
●	51c5a0bb43a9e4b6	WWN	Default
●	51c5a0bb43a9dd87	WWN	Default

Figure 11 - Active/ Active High Availability

On controller A, two floating IP addresses in different subnets are configured and on controller B, another two floating IP addresses of the same 2 subnets are configured. This is a properly load balanced configuration.

SAN Configuration

On the IntelliFlash system, you grant LUN access to the ESXi hosts and their FC/iSCSI ports by mapping the FC/iSCSI initiator groups with the system target groups.

Presenting LUNs to hosts from the IntelliFlash system is called LUN mapping. Before configuring LUN mapping, you should first configure the FC/iSCSI host initiator groups on the system under Services > SAN > Initiator Groups. With the initiator groups configured, LUN mapping can be configured from the Project or LUN level under Provision > Project > Manage and clicking Edit after highlighting a Project or a LUN.

The hosts with FC WWPNs or iSCSI Initiator names in the same initiator group mapped with FC/iSCSI ports in the IntelliFlash target group have access to the same set of LUNs.

iSCSI SAN Configuration

You can configure the iSCSI initiator group and target group through the IntelliFlash Web UI under Services > SAN > iSCSI.

In a non-boot-from-SAN environment, the ESXi hosts in the same cluster should usually have access to the same set of LUNs. In this case, it’s recommended that you create one iSCSI initiator group, and then add the iSCSI initiator node names of those hosts to the initiator group.

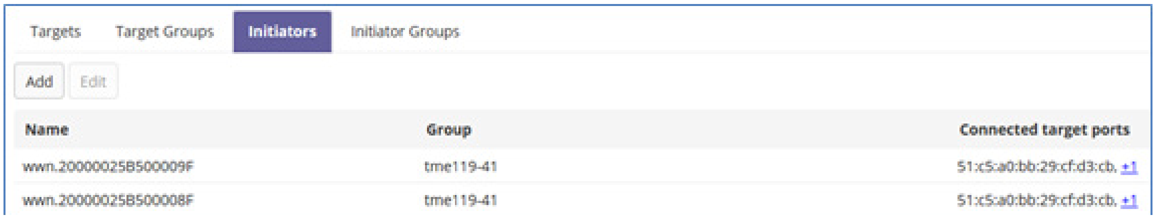
In a boot-from-SAN environment, each ESXi host needs exclusive access to its own boot LUN, and then have the access to the same set of non-booting LUNs. In this case, it’s recommended that you create an iSCSI initiator group for each host, and add the iSCSI initiator name of the particular host only to the initiator group.

The iSCSI target group configuration will be covered, along with LUN Mapping best practices, in the IntelliFlash Storage Provisioning section.

FC SAN Configuration

You can configure the FC initiator group and target group through the IntelliFlash Web UI under Services > SAN > FC.

In a non-boot-from-SAN environment, ESXi hosts in the same cluster should have access to the same set of LUNs. As with iSCSI it’s recommended that you place the WWPNs of those hosts in one FC initiator group as shown in Figure 12.



Name	Group	Connected target ports
wwn.20000025B500009F	tme119-41	51:c5:a0:bb:29:cf:d3:cb, +1
wwn.20000025B500008F	tme119-41	51:c5:a0:bb:29:cf:d3:cb, +1

Figure 12 - FC Initiators

In a Boot-From-SAN environment, each ESXi host needs to exclusively access its own boot LUN and also have access to the same set of non-booting LUNs. Again it’s recommended that you create an FC initiator group for each host, and add the WWPNs of a particular host only to its respective initiator group.

The IntelliFlash system creates a pre-configured default NPIV FC target group when a Pool is created. Each Pool has its own target group. Tintri recommends that you use the default NPIV FC target group for LUN mapping configuration.

With FC initiator and target groups properly configured, you can configure LUN mapping during storage provisioning. For more information, please refer to the LUN mapping best practices in the IntelliFlash Storage Provisioning section.

Storage Provisioning

Pool Provision

On IntelliFlash systems, a pool is a collection of physical disks configured with underlying data redundancy levels. A pool provides storage space to shares and LUNs. All the LUNs and shares in a pool have the same RAID level and share metadata and cache. A pool that contains only NVMe™-SSDs or SAS-SSDs are called an NVMe flash pool or an all-flash pool respectively. A pool that contains both SSD and HDD is called a hybrid pool.

Creating a pool is very straight forward with the pool configuration wizard, which automatically creates and configures pools that are optimized for your capacity and performance requirements. The wizard is available under Provision > Pools > New.

Flow Control

On IntelliFlash systems, a Project is a logical container that contains related LUNs and/or shares in a Pool. A Pool can have multiple Projects. The LUNs and/or Shares in a given Project are usually used to store related data.

For example, one Project might contain the NFS datastores for the VMs of a certain department, and another Project might contain the OS, Data, and Log file LUNs for an SQL server VM.

You can create and configure a Project through the IntelliFlash Web UI under Provision > Projects. Apply the following best known methods when configuring a project:

- Create separate Projects for VMs that require different backup, snapshot, and/or replication schedules.
- Each Project should contain only the LUNs/Shares for one application, or for the VMs that are in the same backup schedule.
- For VMs running IO-intensive applications, SQL Server or Exchange Server for example, it's recommended that you create a dedicated Project for the VM. For example, for an SQL Server VM:
    - o Create a VMware-SQL Project with purpose set to Generic
    - o Create a LUN for VM configuration files and VMDKs for the Windows Server OS itself and the SQL Server software configured with purpose set to Virtual Server.
    - o Create LUNs for the VMDKs supporting the SQL Server database and logs using the appropriate SQL purpose for the type of data they'll store.
- Use the Project creation wizard to create LUNs/Shares for specific applications. The wizard will automatically configure the optimal LUN/share parameters, and pre-configure snapshot schedules for the selected applications, for example, Virtual Server, SQL Server, Exchange, etc.

Note: If you choose the purpose of SQL Server or Exchange for a new Project, it will not let you leverage NFS or create "virtual server" purpose LUNs or shares. If you're planning to use NFS, and/or put the configuration files and OS VMDKs in the same Project, then use a purpose of Generic or Virtual Server for the Project, and then manually create the LUNs/shares using settings outlined in Table 2.

LUN Mapping Configuration

For non-Boot From SAN (BFS) LUNs, Tintri recommends that you configure LUN mapping(s) at the Project level. All the LUNs inside a Project inherit the Project LUN mapping configuration by default, so this simplifies provisioning. This is most easily done by configuring one LUN mapping entry with the Initiator Group containing all the ESXi hosts in the cluster mapped to the default Target Group for the Pool.

For Boot From SAN LUNs, Tintri recommends that you manually remove any mappings specified during Project creation from the Project itself under Provision > Projects > Manage > Access. Then after LUN creation, edit the LUN Access under LUNs > Manage > Access, remove any existing mappings, add only the appropriate Initiator Group containing the one ESXi host, specify LUN ID 0, and choose the default Target Group for the Pool.

iSCSI LUN Mapping

Before creating the first iSCSI Project on an IntelliFlash system, it's recommended to verify the correct floating IPs are bound to the default iSCSI target(s) under Services > SAN.

When adding iSCSI Mapping to Projects or LUNs, it is recommended to always choose default Target Group for the pool, and to always create/choose appropriate Initiator Groups. Never use the All option for Initiator selection unless you're certain of the implications as that can result in data corruption if non-ESXi hosts have access to system.

Fibre Channel LUN Mapping

When creating an FC Project, the LUN Mapping can be configured using the FC Target group and FC Initiator group(s), created earlier during FC SAN configuration.

Modifying LUN Mapping

During Project creation, you're only able to add one LUN mapping with one FC or iSCSI Target Group and one Initiator Group. Additional LUN mappings can be added afterwards through Provision > Project > Manage > Access. By default, all the LUNs in the Project will inherit the Project LUN mapping configuration.

LUN and Share Configuration

Tintri recommends you use the Project creation wizard through the IntelliFlash Web UI to assist in the LUN/share creation for specific applications. Additional LUNs/shares can be added through the IntelliFlash Web GUI via Provision > Project > LUN/shares.

The following table lists shows the recommended properties for the LUNs or shares created for specific applications. It's essentially the same for all applications however the advantage of using the Project wizards is the management time saved configuring multiple LUNs/Shares.

VLAN

When you share a physical network interface for multiple VLANs, for example two VLANs: one for NFS and the other for iSCSI, you can segment the physical port into multiple VLAN ports by adding VLAN links under the designated physical network interface. You can add VLAN links through the IntelliFlash Web UI under Settings > Network > Interface.

Interface Group

It's recommended that you add an Interface group for each storage subnet, and bind the designated physical network interfaces, or VLAN links to it. You should designate a minimum of two network ports per controller for your data network. This ensures the host can continue to access the storage devices on the system in the case of a network path failure.

It is possible to designate only one network interface per Interface group but it will not have any redundancy on that controller.

Floating IP Address

Floating IP addresses are used by the ESXi hosts to mount NFS shares or to access iSCSI LUNs. This ensures the host can continue to access the storage devices on the system in the case of controller failover.

For data networks, when using a single Pool configuration (Controller A only), you should configure the floating IP addresses on controller A only; in the case of 2 Pool configuration (Controllers A/B), an equal number of floating IP addresses should be configured on both the controllers.

Floating IP address can be configured through the IntelliFlash Web GUI under Settings > Network > Interface > Interface Groups or under Settings > High Availability. For each Interface group, it's recommended that you add the same number of floating IP addresses as the number of physical or VLAN interfaces within the Interface group. This helps balance the storage I/O load across the available network paths.

Figure 11 shows and example of an Active/Active configuration:

IntelliFlash LUN/share Parameters				Windows NTFS
Application	**Block Size**	**Compression[1]**	**Deduplication**	
Virtual Server	32KB	**LZ4**	On	32KB
VDI (VMware/Hyper-V)	32KB	**LZ4**	On	32KB
SQL Server (all LUNs)	64KB	LZ4	Off	Yes
Exchange Server: Data	32KB	LZ4	On	32KB[2]
Exchange Server: Log	64KB	**LZ4**	Off	64KB
Oracle: Data	64KB	**LZ4**	Off	N/A
Oracle: Log	64KB	LZ4	Off	N/A

Table 2 - LUN/share Properties Per Application

Notes:

1. The compression is selected automatically and shows as Optimal Performancee
2. It's okay to use either 32KB or 64KB as the NTFS allocation unit size for Exchange Server data volumes. There has not been any noticeable difference between them. In either case, the LUN/share block size should stay as 32KB.

The above recommendations serve as the starting point to configure LUNs and file shares on the IntelliFlash system for the commonly used application and data types. If you have any doubts, the safest and easiest purpose is Virtual Server. Simply edit the LUN/share Settings and disable deduplication if it will be hosting any type of database files, images, or video that will not benefit from it. If needed, please contact your IntelliFlash representatives for further tuning in your specific environment.

Modify LUN Mapping in LUN Properties

The LUN Mapping configuration can be modified at each LUN level through IntelliFlash Web UI, under Manage > Access.

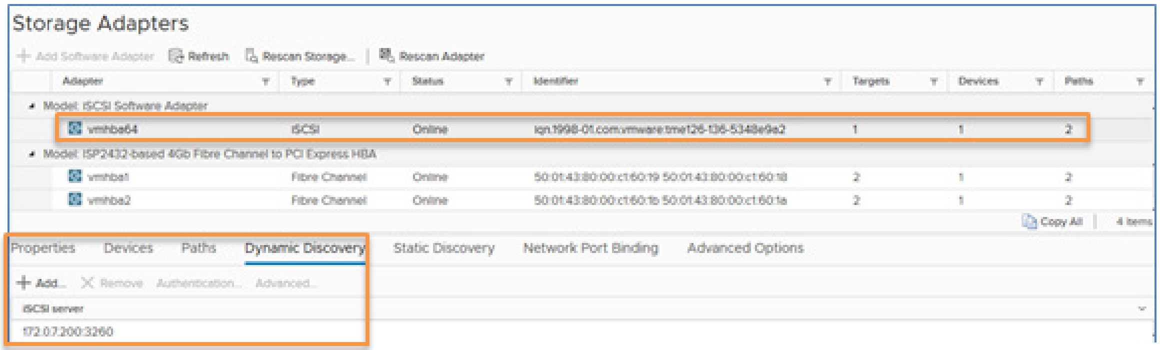
As state earlier, for the BFS LUNs, you will need to modify the LUN mapping configuration at the LUN level to make sure the LUN is only mapped to the intended host with LUN 0.

Storage Configuration on ESXi

iSCSI Discovery

If iSCSI is used as the storage interconnect you must connect to the IntelliFlash target through the iSCSI adapter on the ESXi host. On the VMware vSphere web client, go to Configure > Storage > Storage Adapters > Dynamic Discovery > Add.

Enter one of the floating IP addresses that are bound to the IntelliFlash controller's iSCSI target group and click on OK.



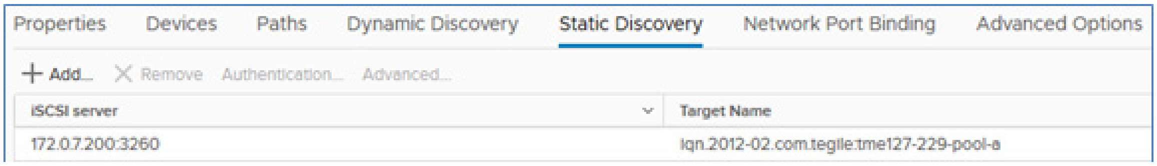
Storage Adapters

Adapter	Type	Status	Identifier	Targets	Devices	Paths
Model: iSCSI Software Adapter						
vmhba64	iSCSI	Online	iqn.1998-01.com.vmware:tme126-136-5348e9a2	1	1	2
Model: ISP2432-based 4Gb Fibre Channel to PCI Express HBA						
vmhba1	Fibre Channel	Online	50:01:43:80:00:c1:60:19 50:01:43:80:00:c1:60:18	2	1	2
vmhba2	Fibre Channel	Online	50:01:43:80:00:c1:60:1b 50:01:43:80:00:c1:60:1a	2	1	2

iSCSI server
172.0.7.200:3260

Figure 13 - Adding iSCSI Target Through Dynamic Discovery

Once the iSCSI Dynamic target is added successfully, click on the Configure > Storage > Storage Adapters > Static Discovery tab to ensure the IntelliFlash iSCSI target was added successfully.



iSCSI server	Target Name
172.0.7.200:3260	iqn.2012-02.com.tegile:tme127-229-pool-a

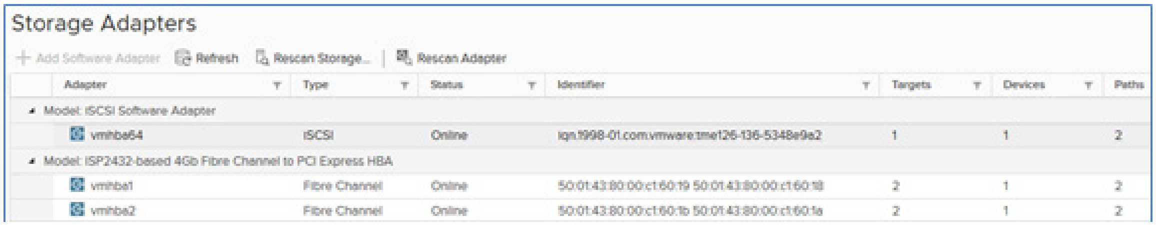
Figure 14 - Static Targets Confirmed

After confirming the static target was added, rescan the iSCSI adapter for new storage devices.

Modifying LUN Mapping

Storage Device Discovery

IntelliFlash FC and iSCSI LUNs are listed under the VMware vCenter or VMware vSphere Client Configure > Storage > Storage Devices list. ESXi automatically discovers new FC or iSCSI LUNs after a short time. You can also manually trigger discovery of new storage devices by clicking Configure > Storage > Storage Adapters > Rescan Storage.



Storage Adapters

Adapter	Type	Status	Identifier	Targets	Devices	Paths
Model: iSCSI Software Adapter						
vmhba64	iSCSI	Online	iqn.1998-01.com.vmware:tme126-136-5348e9a2	1	1	2
Model: ISP2432-based 4Gb Fibre Channel to PCI Express HBA						
vmhba1	Fibre Channel	Online	50:01:43:80:00:c1:60:19 50:01:43:80:00:c1:60:18	2	1	2
vmhba2	Fibre Channel	Online	50:01:43:80:00:c1:60:1b 50:01:43:80:00:c1:60:1a	2	1	2

Figure 15 - Rescan Storage

Storage Device Path Information

Check the storage device path information on each IntelliFlash LUN to make sure the multipathing policy is configured properly for FC or iSCSI, and all the available paths are discovered properly.

Storage Device Multipathing Policy

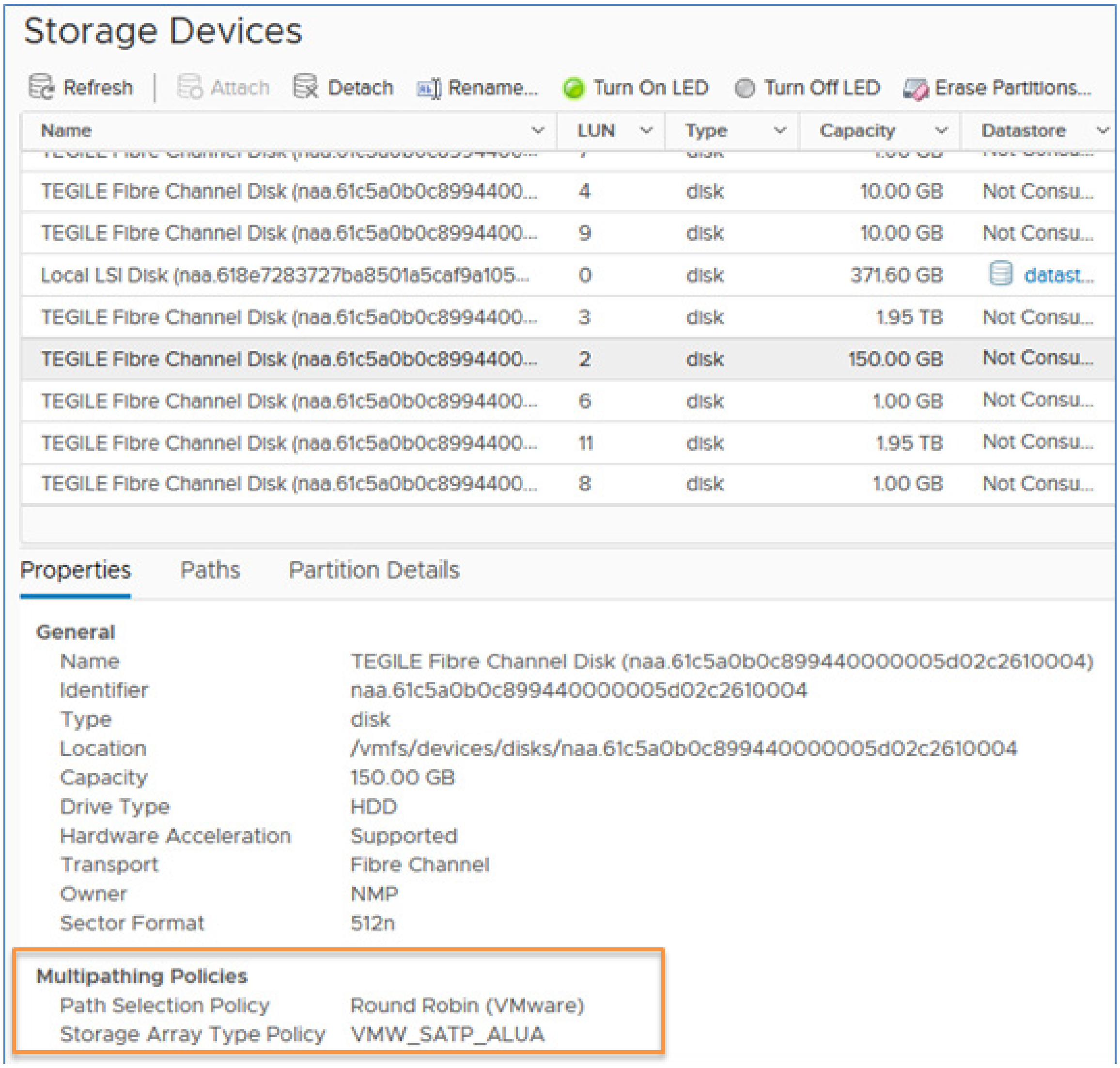
Make sure the multipathing policies on the IntelliFlash LUNs are properly configured as shown in the following table:

Multipathing Policy	Protocol	
	FC	iSCSI
SATP	VMW_SATP_ALUA	VMW_SATP_ALUA
Path Selection	Round Robin	Round Robin

Table 3 - VMware Multipathing Policies

If the multipathing policy on the FC or iSCSI LUN is not consistent with the above table, it means the multipathing rules have not been configured properly on the host. Please consult with WD IntelliFlash support to acquire and use the IntelliFlash VMware Best Practices PowerCLI script to verify and apply the SATP (Storage System Type Plugin) rules. These SATP rules automatically configure multipathing for all IntelliFlash LUNs.

To check the multipathing policy, select an IntelliFlash FC or iSCSI LUN, and click Properties. The path selection is shown as Round Robin (VMware) and the storage system type is shown as VMW_SATP_ALUA.



Name	LUN	Type	Capacity	Datastore
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	4	disk	10.00 GB	Not Consu...
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	9	disk	10.00 GB	Not Consu...
Local LSI Disk (naa.618e7283727ba8501a5caf9a105...	0	disk	371.60 GB	datast...
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	3	disk	1.95 TB	Not Consu...
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	2	disk	150.00 GB	Not Consu...
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	6	disk	1.00 GB	Not Consu...
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	11	disk	1.95 TB	Not Consu...
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	8	disk	1.00 GB	Not Consu...

General

Name	TEGILE Fibre Channel Disk (naa.61c5a0b0c899440000005d02c2610004)
Identifier	naa.61c5a0b0c899440000005d02c2610004
Type	disk
Location	/vmfs/devices/disks/naa.61c5a0b0c899440000005d02c2610004
Capacity	150.00 GB
Drive Type	HDD
Hardware Acceleration	Supported
Transport	Fibre Channel
Owner	NMP
Sector Format	512n

Multipathing Policies

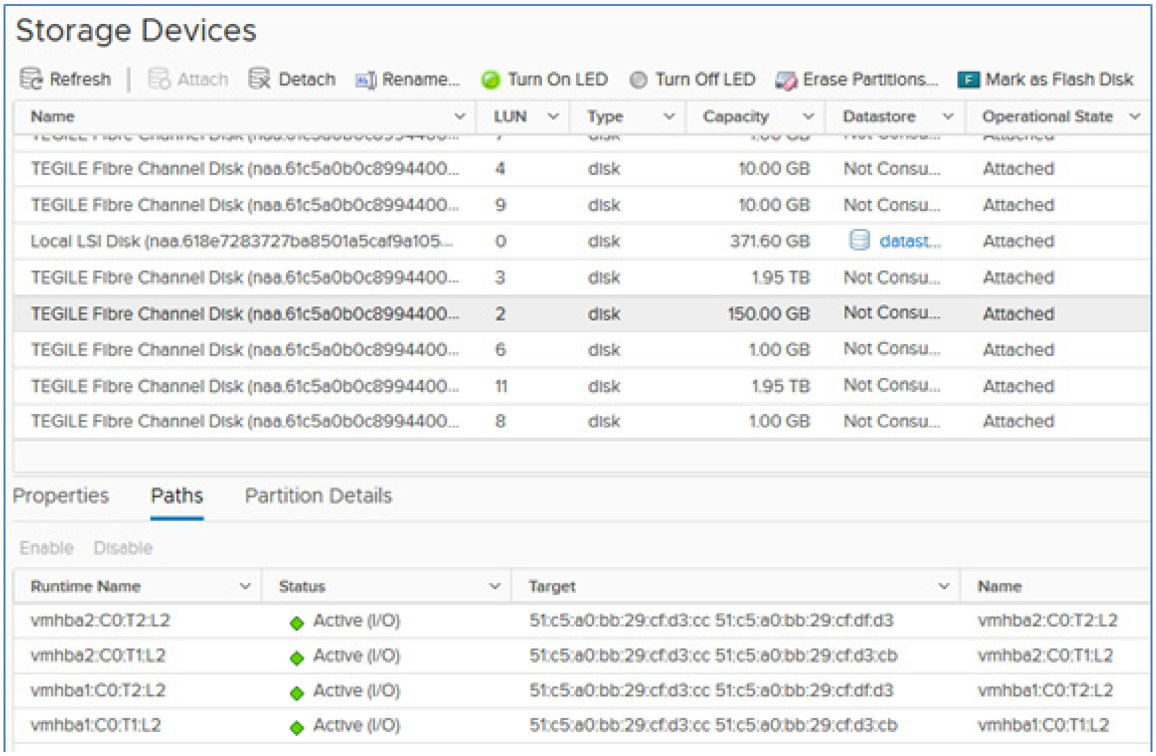
Path Selection Policy	Round Robin (VMware)
Storage Array Type Policy	VMW_SATP_ALUA

Figure 16 - Properly Configured Multipathing Policies

Clicking on Paths displays all the paths from the device to host. Figure 17 provides an example on the multiple paths to an IntelliFlash FC LUN.

When using NPIV for FC all the paths will show as Active (I/O) because the NPIV WWPNs move with the Pool as a resource. The ESXi host sees the same WWPNs no matter what controller is managing the Pool resource. In the case of Figure 17, there are 2 FC ports on ESXi each zoned in a single fabric to each of 2 NPIV ports on the IntelliFlash system for a total of 4 paths.

Somewhat more common when using NPIV on the IntelliFlash system would be two separate fabrics, such that each host FC port is zoned to just one NPIV port per pool, which would give you 2 total paths per LUN.



Name	LUN	Type	Capacity	Datastore	Operational State
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	4	disk	10.00 GB	Not Consu...	Attached
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	9	disk	10.00 GB	Not Consu...	Attached
Local LSI Disk (naa.618e7283727ba8501a5caf9a105...	0	disk	371.60 GB	datast...	Attached
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	3	disk	1.95 TB	Not Consu...	Attached
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	2	disk	150.00 GB	Not Consu...	Attached
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	6	disk	1.00 GB	Not Consu...	Attached
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	11	disk	1.95 TB	Not Consu...	Attached
TEGILE Fibre Channel Disk (naa.61c5a0b0c8994400...	8	disk	1.00 GB	Not Consu...	Attached

Runtime Name	Status	Target	Name
vmhba2:C0:T2:L2	Active (I/O)	51:c5:a0:bb:29:cf:d3:cc 51:c5:a0:bb:29:cf:df:d3	vmhba2:C0:T2:L2
vmhba2:C0:T1:L2	Active (I/O)	51:c5:a0:bb:29:cf:d3:cc 51:c5:a0:bb:29:cf:d3:cb	vmhba2:C0:T1:L2
vmhba1:C0:T2:L2	Active (I/O)	51:c5:a0:bb:29:cf:d3:cc 51:c5:a0:bb:29:cf:df:d3	vmhba1:C0:T2:L2
vmhba1:C0:T1:L2	Active (I/O)	51:c5:a0:bb:29:cf:d3:cc 51:c5:a0:bb:29:cf:d3:cb	vmhba1:C0:T1:L2

Figure 17 – LUN Paths with Proper Path Selection Policy

Note: If using the physical FC ports (not recommended) on an IntelliFlash system and using a similar type of zoning then there would be 8 total paths. However, 4 paths would show as Active (I/O) but 4 paths would show as Stand by.

For an IntelliFlash iSCSI LUN the path information would look similar to the NPIV FC LUN example, with the Path Selection showing as Round Robin (VMware) and the Storage System Type as VMW_SATP_ALUA.

VMware Datastores

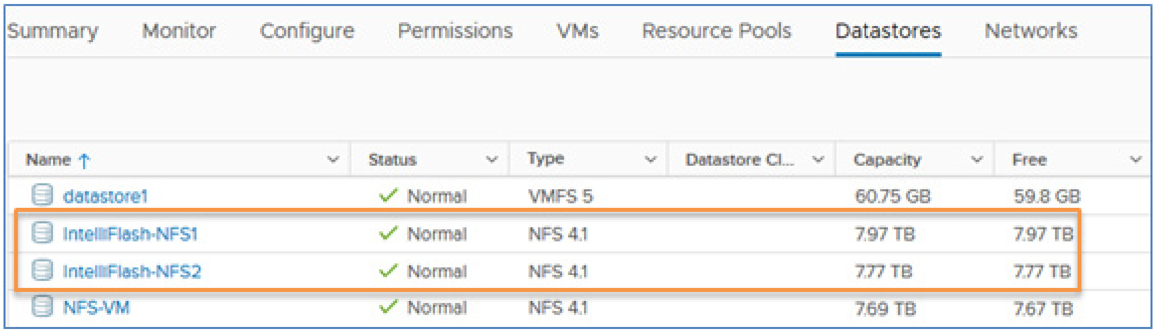
NFS Datastores

Before adding any NFS datastores to ESXi, you must validate that IntelliFlash Best Practice settings are applied. Please consult with WD IntelliFlash support to get and use the IntelliFlash VMware Best Practices PowerCLI script to verify and apply the Best Practice Settings for IntelliFlash NFS datastores, which includes the VAAI host extension to provide hardware acceleration.

When adding NFS datastores, if you have multiple floating IPs per pool on the NFS subnet, alternate the use of those IPs for any given datastore. This helps in evenly distributing the I/O load across the multiple network paths, but you must make certain to use the same IP for any given datastore on all hosts in order for HA and vMotion to function as expected.

To add an NFS datastore from within vCenter simply right-click the Datacenter, Cluster, Host Folder, or even individual ESXi host, select Storage > New Datastore, and complete the wizard.

Note: Tintri highly recommends NFSv3 for all IntelliFlash datastores today. The IntelliFlash system supports version 4.1 if you're running IntelliFlash version 3.10.0.0 or higher and ESXi 6.7u1 or higher, but the ESXi NFS client generally offers higher performance and better stability with NFS version 3.



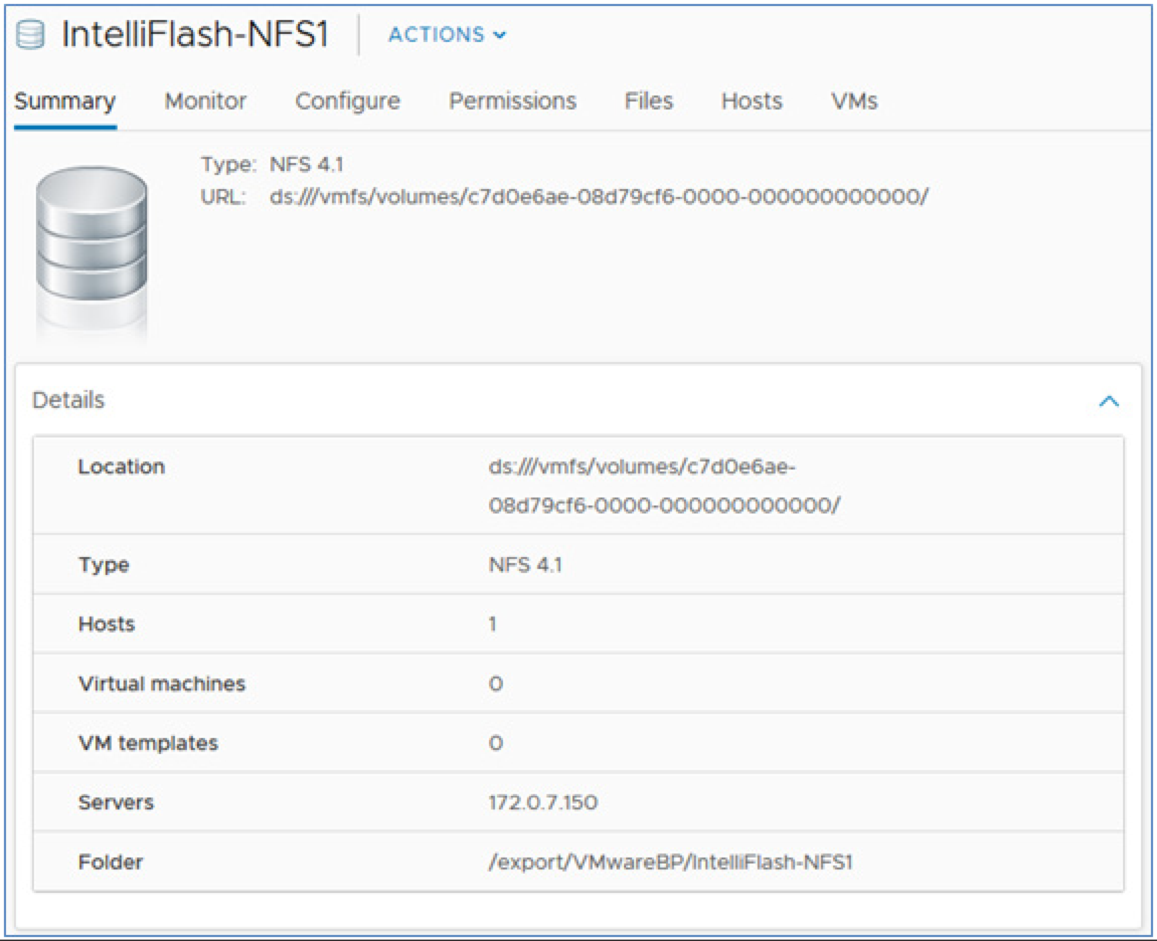
Name ↑	Status	Type	Datastore Cl...	Capacity	Free
datastore1	✓ Normal	VMFS 5		60.75 GB	59.8 GB
IntelliFlash-NFS1	✓ Normal	NFS 4.1		7.97 TB	7.97 TB
IntelliFlash-NFS2	✓ Normal	NFS 4.1		7.77 TB	7.77 TB
NFS-VM	✓ Normal	NFS 4.1		7.69 TB	7.67 TB

Figure 18 – Multiple NFS Datastores Load Balancing

Figure 18 shows an example of mounting two NFS shares which are using different floating IP addresses on the IntelliFlash system. The first NFS share is mounted through a floating IP address (172.0.7.150) and the second NFS share is mounted through a second floating IP address (172.0.7.151). Clicking on the datastore opens up details on the IntelliFlash NFS device backing the datastore as shown in Figure 19.

Both floating IP addresses are associated with the same IntelliFlash Interface Group with two 10GbE physical network interfaces per system controller. The I/O to the first share are effectively sent through the first 10GbE port, and the second NFS share effectively uses the second 10GbE port on the system. In this way, it achieves load balance across these two NFS shares.

A more common and recommended configuration involves leveraging LACP link aggregates if your 10GbE switches can support them, enabling L2, L3, & L4 hashing on the IntelliFlash system, and provisioning a single IP for NFS per IntelliFlash pool. This can simplify provisioning, and provide I/O distribution across the 10GbE interfaces as multiple ESXi hosts connect. Please consult with your IntelliFlash SE and/or the IntelliFlash Professional Services team for detailed networking best known methods.



Type: NFS 4.1
URL: ds:///vmfs/volumes/c7d0e6ae-08d79cf6-0000-000000000000/

Details

Location	ds:///vmfs/volumes/c7d0e6ae-08d79cf6-0000-000000000000/
Type	NFS 4.1
Hosts	1
Virtual machines	0
VM templates	0
Servers	172.0.7.150
Folder	/export/VMwareBP/IntelliFlash-NFS1

Figure 19 – IntelliFlash System Details Backing NFS Datastore

FC or iSCSI Datastores

To add a disk/LUN type datastore using an IntelliFlash FC or iSCSI LUN, simply follow the datastore creation wizard on the VMware vCenter web client by right-clicking on the Datacenter, Cluster, Host Folder or even individual ESXi host and selecting Storage > New Datastore > VMFS.

Using VMFS version 6 provides automatic space reclamation support which is discussed in the Thin Provisioning Primitives section.

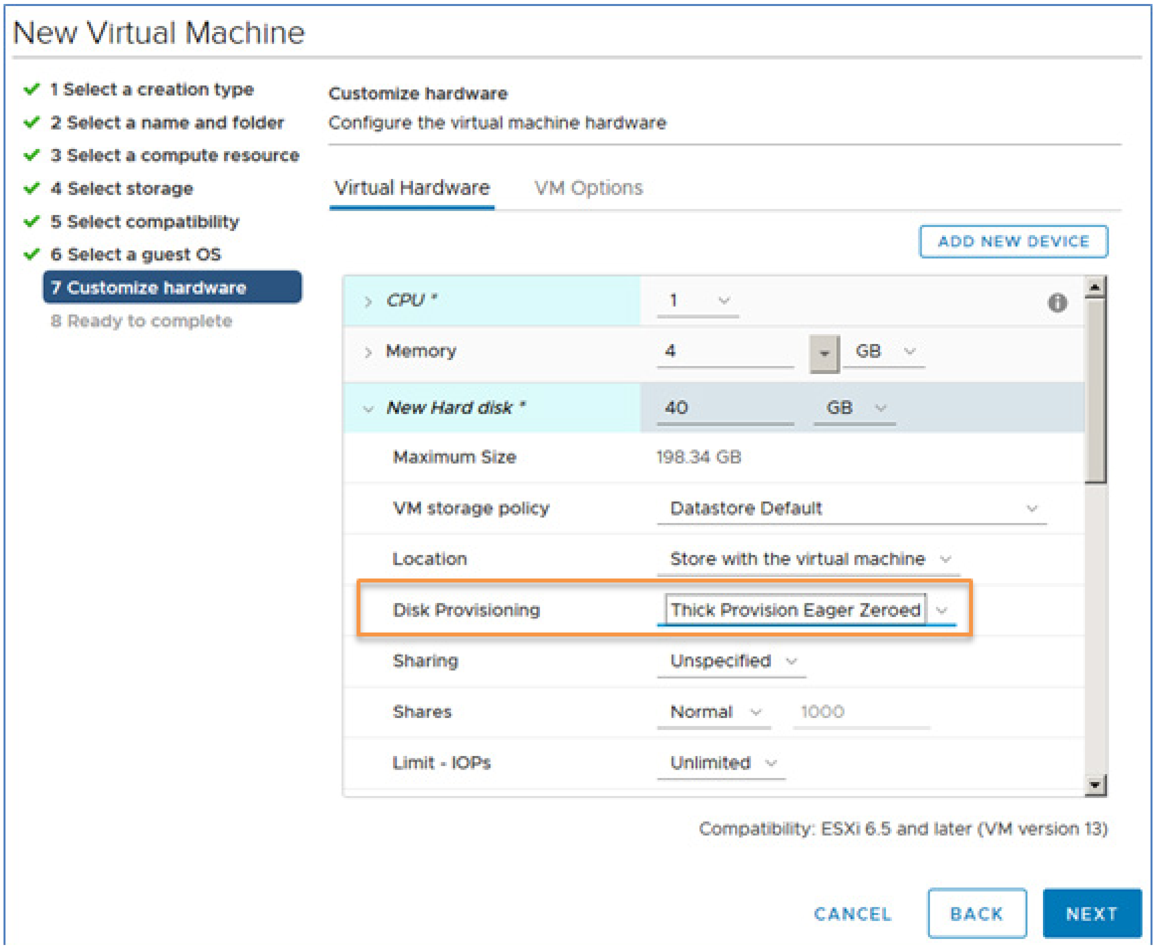
VM Deployment and Maintenance

Virtual Disk Provision Type

Provisioning iSCSI or FC Virtual Disks

When you create a new VM, or add a virtual disk (VMDK) to an existing VM on an IntelliFlash iSCSI or FC datastore there are a couple of paths to consider. Older versions of ESXi did not have in-guest UNMAP capabilities so picking thin provisioning did not have much appeal.

The upfront performance savings from Thick Provision Eager Zeroed (Figure 20) virtual disks seemed appropriate. With this type of provisioning the entire disk is zeroed upfront and as opposed to lazy zeroed or thin provisioning. This takes slightly longer up front but gives a tiny edge in performance later on because I/O won't have to wait for blocks to be zeroed.



Customize hardware
Configure the virtual machine hardware

CPU *: 1
Memory: 4 GB
New Hard disk *: 40 GB
Maximum Size: 198.34 GB
VM storage policy: Datastore Default
Location: Store with the virtual machine
Disk Provisioning: Thick Provision Eager Zeroed
Sharing: Unspecified
Shares: Normal 1000
Limit - IOPs: Unlimited

Compatibility: ESXi 6.5 and later (VM version 13)

Figure 20 – Thick Provision Eager Zeroed Virtual Disk

However, with ESXi 6.0 for Windows and ESXi 6.5 for Linux there is in-guest UNMAP/TRIM support. Now thin provisioning can be maintained end-to-end and the argument for a tiny improvement in performance pales when compared to better space savings that Thin Provisioned virtual disks can provide.

See the IntelliFlash and VAAI section for more information on the Block Zero primitive and section Thin Provisioning Primitives for more on in UNMAP and thin provisioned virtual disks.

Provisioning NFS Virtual Disks

When creating a virtual disk on an IntelliFlash NFS share, it is recommended to always set the disk provisioning to Thin Provision. If the recommended IntelliFlash VAAI host extension for NFS is installed, then it will allow you to select Thick, but this will provide zero benefit on IntelliFlash NFS datastores.

File System Block Alignment on Guest Operating Systems

IntelliFlash systems perform best when the file system partition and file system block size are aligned with the block size on the IntelliFlash LUN/share.

In a virtual environment, it's recommended you consider block alignment end-to-end. This includes everything from the storage device block size to the hypervisor file system offset and block size, as well as the partition offset and file system allocation unit size on the guest operating system.

VMware VMFS version 5 which is used in ESXi 5 and VMFS version 6 introduced in ESXi 6.5, use a file system offset of 1MB and a block size of 1MB, which align with all IntelliFlash supported LUN/share block sizes so there are no block alignment concerns on VMFS.

In VMware ESXi based virtual environments the key to block alignment is to align the guest OS's file system offset and allocation unit size to the IntelliFlash LUN/share block size.

Adjust Windows NTFS File System Offset

On Windows 2008 and later, the default MBR partition offset is 1MB, which aligns perfectly with the IntelliFlash system. So when creating a partition simply keep the Windows default.

Adjust Windows NTFS File System Allocation Unit Size

All Windows NTFS file systems use 4KB as the default allocation unit size. For best performance, it's recommended that you match the NTFS file system allocation unit size with the IntelliFlash LUN/share block size.

For example, if the block size of your LUN or share is 32KB, then set the NTFS file system allocation unit size to 32KB. If the block size of your LUN or Share is 64KB, then set the NTFS file system allocation unit size as 64KB.

You may modify the NTFS file system allocation unit size through the diskpart utility or through Windows Disk Manager on the guest operating system.

Block Alignment at Scale

In an environment with many VMs to be provisioned, it can become time consuming to apply block alignment practices on many individual VMs.

To apply at scale, it's recommended that you apply the block alignment best methods described above on a few gold VM templates of the required versions of Windows and Linux®, and then create additional VMs by cloning from these templates. In this way the block alignment configuration can be automatically inherited by the new VMs cloned from the templates without repeating the steps manually.

In addition, IntelliFlash systems support VAAI both for block and file. It offloads the VM clone file copy operation from the server and network, and reduces the time to clone those VMs.

Block Alignment on Legacy VMs

When VMs are migrated from legacy storage, chances are that the blocks will not be aligned. You can use Storage vMotion or cross-host Storage VMotion to transform virtual disks.

Contact your Tintri representative to help you craft a migration plan, and to apply the block alignment best methods accordingly.

Additional VM Creation Best Practices

VM NIC Configuration

For performance considerations, it's recommended that you choose the network adapter type of VMXNET 3 for any 10GbE virtual network interface in a VM.

The network adapter driver for VMXNET 3 is built-in to some modern operating systems, but also is included in VMware Tools. When a guest operating system without the native VMXNET 3 driver is initially installed, the VMXNET 3 network adapter will not be automatically recognized. After VMware Tools installation, the VMXNET 3 adapter will be recognized.

Virtual SCSI Controller Type

There are two commonly used virtual SCSI controller types:

- LSI Logic SAS
- VMware Paravirtual (aka PVSCSI)

The queue depths for the LSI Logic SAS and PVSCSI (VMware Paravirtual) are shown in Table 4. The LSI Logic controller adapter queue depth is fixed on both Windows and Linux and has a much smaller maximum of 128 per adapter compared to 256 on PVSCSI.

The queue depths on the PVSCSI controller have a much higher default setting and have the ability for much higher maximum values, up to 8 times that of the LSI Logic SAS controller.

Queue	PVSCSI	LSI Logic SAS
Default Adapter Queue Depth	256	128
Maximum Adapter Queue Depth	1024	128
Default Virtual Disk Queue Depth	64	32
Maximum Virtual Disk Queue Depth	254	32

Table 4 – VMware SCSI Controller Queue Depths

Therefore, for VMs that run I/O intensive applications such as a database, email servers or if the application requires higher queue depths, it’s recommended to use the VMware Paravirtual SCSI controller.

Check the Performance Tuning section of this document for instructions on how to change PVSCSI queue depths.

VMware Tools

It’s always recommended that you install VMware Tools on the guest operating system of each VM.

VMware Tools not only includes the vital device drives for VMware virtual adapters like VMXNET3, but also the VSS providers for VMware virtual disks, which can optionally be used by the IntelliFlash system to create VM filesystem consistent (quiesced) snapshots for backup and recovery.

When you deploy VMs at scale, it’s a best method to create a gold template VM for each of the selected guest operating systems, and install VMware Tools on the gold template VMs. This way, cloning from the gold templates can create additional VMs that automatically include VMware Tools.

VM Virtual Disk Timeout

Recommended best practice for the disk timeout value on the guest OS is 180 seconds in order to ensure the VM can continue to run properly after storage controller failover and failback.

Configure the disk timeout value after VMware Tools has been installed because VM Tools modifies the disk timeout on Linux VMs for VMware virtual disks. On a Windows VM, modify the value of the following Windows registry key to 180:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Disk\TimeOutValue
```

On a Linux VM, add an udev rule with the following entry

```
DRIVERS==”sd”, SYSFS{TYPE}==”0|7|14”, RUN+=”/bin/sh -c ‘echo 180 > /sys$$DEVPATH/timeout’”
```

This setting not only applies to the guest OS virtual disk but to all disks on a VM including VMware virtual disks, RDMs, or in-guest iSCSI disks.

Jumbo Frames on Guest Operating Systems

If in-guest iSCSI or NFS is used on a guest operating system VM and Jumbo Frames are in use end-to-end for the host storage on the same network, then Jumbo Frames should also be enabled on the network ports within the guest OS.

Note: The Jumbo Frame setting could be lost if the VM is created through a VM clone on certain guest operating systems. After cloning a VM always verify the Jumbo Frames setting on the guest OS of the new VM.

Backup and Restore

This section looks at 2 major aspects of backing up an ESXi environment – VMs and applications.

VM Backup and Restore

The IntelliFlash system supports VM-consistent snapshot-based backup and restore by integrating with VMware vSphere.

VM Tools

VMware vSphere can use VMware Tools on the guest OS to create a VM-consistent snapshot. IntelliFlash systems leverage the same vSphere native snapshot support. Therefore, VMware Tools must be installed and enabled on each VM.

IntelliFlash VM Snapshot Configuration

For VM-consistent backup and restore, apply the following best practices on the IntelliFlash Web UI:

- Add the VMware vCenter Server and make sure to enable Quiesce on the Add vCenter Server screen if a VM-consistent snapshot is required. You can add the vCenter server by going to Settings > App-Aware > VMware Servers.
- Group VMs with the same backup/replication requirements into the same Project(s).
- Schedule the backup/replication schedule at the Project level, and make sure to enable Quiesce if a VM-consistent snapshot is required.

VM Restore

IntelliFlash can leverage system-based snapshots to restore all of the VMs on a LUN/share through snapshot rollback. It also allows you to restore individual VM or individual files on the guest operating system through snapshot clone.

Note: Snapshot rollback cannot be reversed so it is always advisable to start with a snapshot clone to verify data before rolling back (or promoting the clone).

Application Data Backup and Restore

To understand the conclusions and the example at the end of this section several topics around how data can be stored in a VMware ESXi environment will need to be covered.

The IntelliFlash system supports several Windows based applications that can be backed up at the application level. Thus, the focus of this section will be mainly around the Windows operating system.

Options to Store Application Data

There are several different ways a Windows-based application data snapshot is created and it's based on how the underlying storage is configured.

The following table describes the various ways to configure storage to store application data in a VMware environment, and what VSS-based snapshot agent is used for application consistent backup and restore.

Queue	Interconnect Type	Storage Adapter	Vendor / Model	Snapshot Agent
Direct Connect	iSCSI	In-Guest Software iSCSI Adapter	TEGILE[1] INTELLIFLASH	TDPS[2]
Physical RDM	FC, iSCSI	LSI Logic SAS or VMware Paravirtual	TEGILE[1] INTELLIFLASH	TDPS[2]
VMware Virtual Disk	**FC, iSCSI, NFS**	**LSI Logic SAS or VMware Paravirtual**	VMDK[3]	VM Tools VMware VSS HW Provider

Table 5 – Application Consistent Backup For Various Disk Types

Notes:
1. SCSI Vendor ID shows as TEGILE not Tintri
2. Tegile Data Protection Service
3. VMware Virtual Disk

When choosing RDM or VMDK options, for performance consideration, Tintri recommends you use VMware Paravirtual SCSI Adapter for any I/O intensive application running on the VM.

Application Data on RDM or Direct Connect iSCSI

FC and iSCSI RDM and in-guest iSCSI disks can be directly and exclusively accessed by a VM. When using RDM or in-guest iSCSI disks to store large application datasets or I/O intensive applications, such as large SQL Server databases or Exchange Server, you can achieve the following benefits:

- Optimal application performance
- Flexible snapshot backup schedules based on the application needs
- Application-aware, consistent snapshot backup and restore via the Tegile Data Protection Service (TDPS)

To protect application data on these disks, apply these recommendations:

- Configure the LUNs appropriately for the application
- When using RDM or In-guest iSCSI, configure the Windows multipathing
- Download the TDPS installer from the IntelliFlash Web UI, and install TDPS on each Windows guest
- Add the Windows guest as a Windows Server under Settings > App-Aware > Windows Servers in the IntelliFlash Web UI
- Ensure the IntelliFlash UI properly displays the TDPS version and the number of LUNs presented to the guest OS. If TDPS is configured properly it should match the LUNs that are presented and detected on the virtual Windows server.

Application Data on VMware Virtual Disk

The application data can also be stored on VMware Virtual Disk (VMDK), and a VMDK can reside on either an FC or iSCSI LUN or on an NFS share.

In order to achieve the optimal application performance and allow flexible snapshot schedules for large application datasets or I/O intensive applications Tintri recommends that you allocate separate LUNs or shares for application data and then separate LUNs or shares for the VM configuration files and OS VMDK.

Using the VMware VSS HW Provider in the VM Tools will only create VM-Level snapshots. VMware VSS HW Provider is a generic VSS HW provider and is not able to do additional application-aware operations right before or after snapshot creation like truncating application logs.

For better application-consistent backup and restore support when using VMware Virtual Disk to store application data, it's recommended that you use a third-party backup software, such as Veeam, and leverage a third party VSS requester that supports the application and can perform additional operations to quiesce the VM and ensure all transactions are safely written to disk. Some backup software will also need additional pre-freeze and post-thaw scripts for proper quiescence of the VM.

Note: Only Commvault and Veeam support integration with IntelliFlash snapshots for backup. When using other third-party backup software it will not take advantage of the hardware-based snapshot on the IntelliFlash system.

Separating Application Data from Guest OS

Tintri recommends that you place the VM guest OS VMDK/configuration files and the application data on separate LUNs or Shares. With this configuration you can achieve the following benefits:

- Maximize capacity savings and performance by configuring the LUN/share based on functionality.
  - Guest OS –
    - Deduplication is highly effective when multiple VMs are running similar versions of a guest operating system and application software.
  - Application data –
    - Deduplication isn't as effective, especially for database and media files.
    - I/O request size varies from application to application.
- The separate LUNs and shares can be fine-tuned independently for the guest OS and individual application. This way it maximizes the capacity savings and achieves the optimal performance for both VMs and applications.
- Flexible snapshot backup and replication schedule.
  - o Typically, it's best to keep all VM files together in same Project and schedule snapshots at the Project level. This makes system-based snapshots more useful for VM recovery, otherwise snapshots may be at different times which could make VM unrecoverable or at a minimum much more difficult.
- Reliable application-consistent snapshots, with the IntelliFlash system based snapshots.
  - o Coupled integration with TDPS and Microsoft VSS ensures reliable snapshots.
  - o Strict time requirement from Microsoft VSS mandates several time limits during snapshot creation. When mixing VM and application data on the same LUN, it can potentially increase the time to create a snapshot, as it has to wait for all the VMs and applications on the LUN to quiesce. It could potentially exceed the Microsoft VSS mandated time limit as a result. The snapshot may not be application consistent. When you have dedicated LUNs for an application, the snapshot creation is much faster and more likely to finish within the Microsoft VSS mandated time limit, making it more reliable.

Consider all the recommendations above when provisioning storage for applications running in virtual environments. Additionally, allocating separate LUNs or shares for each application, temporary data, pages files, etc. All of these data types will have their own requirements and would benefit from separation. For example, temporary data doesn't usually require backups or replication thus saving capacity and improving performance.

Application Data Restore

IntelliFlash system enables you to leverage system-based snapshots to restore all the data on a LUN or share through snapshot rollback. It also lets you restore individual files by leveraging snapshot clone.

Note: that snapshot rollback cannot be reversed though, so it is always advisable to start with a snapshot clone to verify data before rolling back (or promoting the clone).

Application Deployment Example

Figure 21 shows an example of deploying an application in a VMware environment applying all of the recommendations previously discussed.

In this example:

- VM
  - o The Windows OS is installed on a VMware virtual disk (VMDK) from a separate IntelliFlash LUN. The LUN is formatted with a block of 32KB, compression is set to lz4 and deduplication is enabled.
  - o The application running on the Windows VM is an SQL Server data warehouse application.
  - o On the Windows guest OS, both VMware Tools and the Tegile Data Protection Service (TDPS) are installed.
  - o VM snapshots are scheduled on the Project that contain the VM LUN. VMware Tools is used to create VM-consistent snapshots.

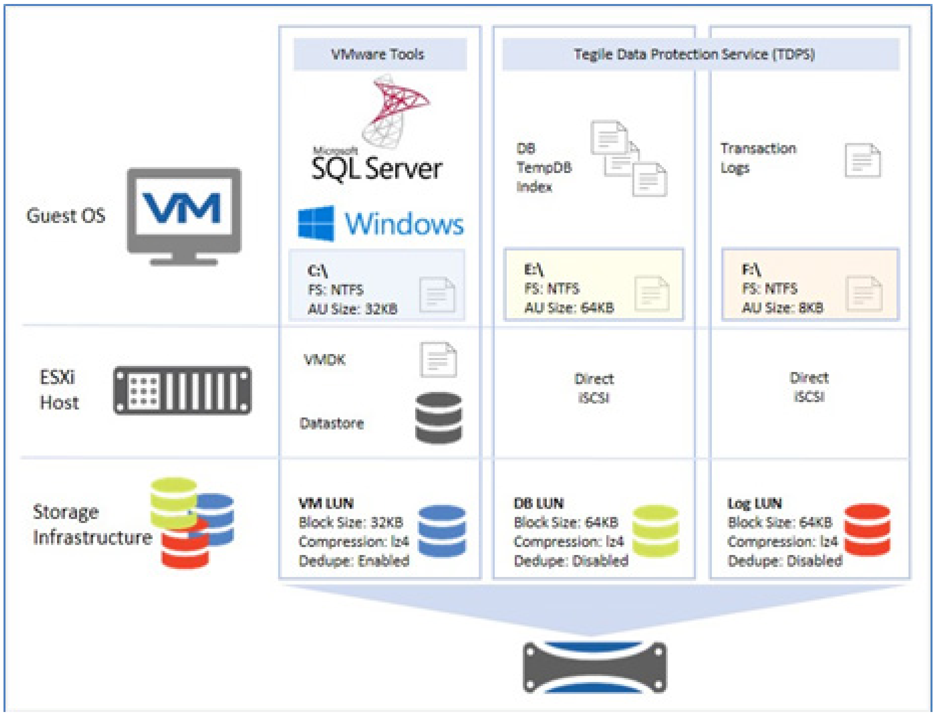


Figure 21 – Application Deployment Example with Microsoft SQL Server

- Application
  - o The SQL Server database and logs are stored on two separate LUNs, configured in the VM as RDM storage devices.
  - o Both LUNs are create with a 64KB block size, compression is set to lz4, and deduplication is disabled on both as database datasets are usually compressible but do not have much duplicated data content.
  - o The database LUN is formatted with NTFS and the block size set to 64KB, which is equal to the average I/O size that SQL Server usually sends to storage in a data warehouse application.
  - o The Log LUN is formatted with NTFS and the block size set to 8KB, which is close to the average I/O size that SQL Server usually sends to the storage in a data warehouse application.
  - o The application data snapshots are scheduled on the Project that contains the database and log LUNs. TDPS is used to create application-consistent snapshots.

Disaster Recovery

IntelliFlash systems support VM and application consistent disaster recovery through snapshot-based remote replication. Through integration with VMware vSphere and Microsoft VSS (Volume Shadow Copy Service), it replicates VM and application-consistent snapshots to the remote system. In this way, the application can gracefully recover using the remote replica on the secondary site in case the primary site is down. Since disaster recovery simply replicates VM and application-consistent snapshots the snapshot-related best practices also apply to remote replication. Please refer to the Backup and Restore section for these best practices.

Additional replication specific best methods are described in the following sub-sections.

Replication Network Consideration

For general network recommendations, please refer to Replication Network sub-section under the Planning Ahead section of this document.

At a minimum, one network interface per controller is required for replication. For path redundancy, Tintri recommends connecting two network interfaces per controller.

Configure Floating IP on the Replication Network

You must configure replication to use a floating IP address. This will ensure the replication can continue functioning in the case of controller failover. As mentioned before use a dedicated network for replication.

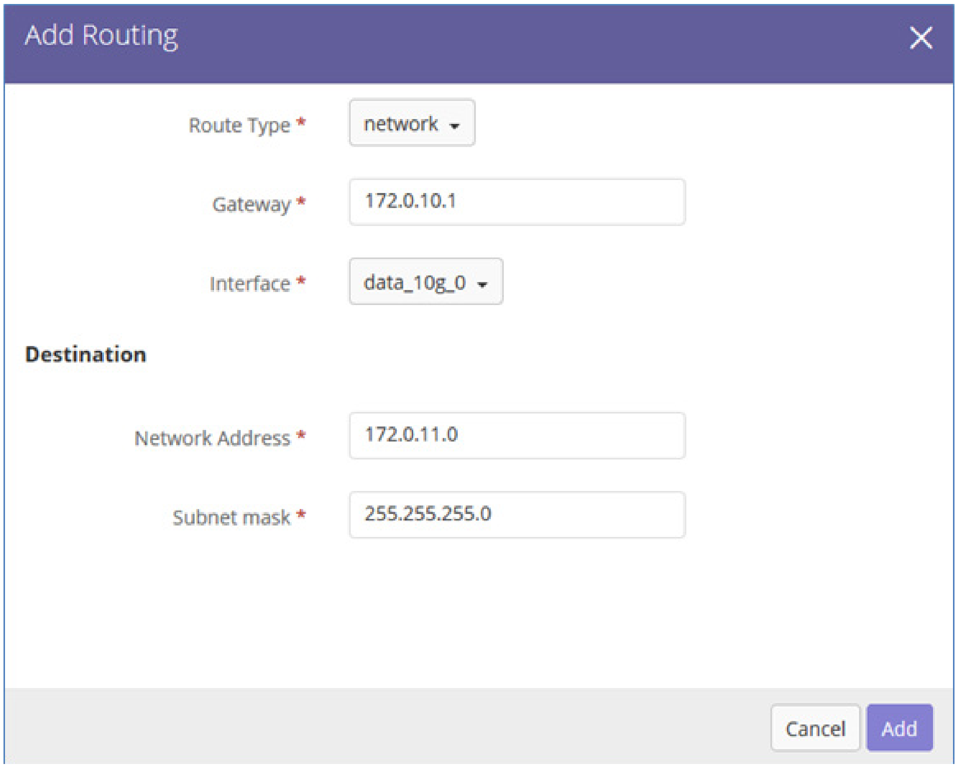
To configure a floating IP on the dedicated replication network:

- Connect the specific physical ports on both controllers to the intended network switch.
- On each controller create an interface group, under Settings > Network > Interface, dedicated to replication and add the physical or VLAN network interfaces into this interface group. If sharing the same physical ports with functions, for example, sharing ports between management and replication
    - o Configure VLANs to segment the networks accordingly on the switch.
    - o Configure the VLAN interfaces on the IntelliFlash system
- On each active controller, add a floating IP, and assign the IP address and netmask accordingly by using the replication interface group.

Configure Routing for the Replication Network

For each destination replication network, add a static network route under Settings > Network > Advanced through the IntelliFlash Web UI. This ensures the replication traffic is routed through the designated gateway instead of through the default gateway on the management network.

For example, if the gateway of the replication network is 172.0.10.1 on the primary site and the replication target system is on the 172.0.11.x network at the secondary site, a static route should be created for the destination network of 172.0.11.0 as shown in Figure 22.



Add Routing

Route Type * network

Gateway * 172.0.10.1

Interface * data_10g_0

Destination

Network Address * 172.0.11.0

Subnet mask * 255.255.255.0

Figure 22 – Adding Replication Network Route

Configure DNS or Static Host Table

Typically the initial replication is configured and completed by having both the source and destination systems at the same site then one of the systems is moved to the remote site. After the move, the system's IP address may change but fully qualified domain name should remain the same.

To achieve this simple method for the initial replication it's recommended to use an FQDN to configure the replication target. With the FQDN following the system, replication will continue as soon as the DNS entry or static host table is modified to reflect the IP address change.

Either DNS or static host table can be used to configure the mapping between the fully qualified domain name of the system and the replication floating IP address. Please refer to IntelliFlash User Guide for the instructions about how to configure DNS and the static host table through IntelliFlash Web UI.

When configuring the DNS or static host table, consider the following:

- Ensure the DNS is reachable from both the source and target systems.
- When creating the mapping of an FQDN and a floating IP address use a different host name from the controllers. For example:

Controller Name	t4800-c1.mycompany.com
Replication Floating IP	172.0.10.100
Actual Mapping	rep-t4800-c1.mycompany.com ↔ 172.0.10.100

Figure 23 – Mapping Floating IP to FQDN

Note: Notice the actual mapping used a different host name from the controller's hostname.

Configure Replication

You can configure replication and define replication schedules under Provision > Projects > Manage > Data Protection > Replication through the IntelliFlash Web UI.

When configuring replication, apply the following best methods:

- When specifying the replication target system, use a fully qualified domain name, instead of an IP address
- For a VM-consistent or application-consistent snapshot, check Quiesced
- Designate a replication schedule that avoids overlapping other replication or snapshot schedules

Please refer to the IntelliFlash User Guide to find detailed procedures for configuring replication.

Recovery From Replication

When recovering a virtualized environment, you can clone the desired snapshot on the replica system and make cloned LUNs or shares accessible to the hosts accordingly. Since the cloned LUNs/Shares are based on VM and application-consistent snapshots, the VMs and applications can recover gracefully and instantaneously.

Recovery Using SRM/SRA

In support of VMware Site Recovery Manager (SRM), Tintri provides a Site Recovery Adapter (SRA) specifically designed for IntelliFlash systems that can be plugged into SRM to allow VMware SRM to leverage the IntelliFlash system-based storage replication for seamless VM disaster recovery between datacenters.

Starting from IntelliFlash release 3.5 VMware SRM is supported. You may download the IntelliFlash SRA through the system web UI under Settings > Administration > Plugins.

Before configuring SRM you need to create a replication relationship for the protected data Project between the systems of the protected site (source) and the recovery site (target).

This can be achieved through IntelliFlash web UI under Provision > Projects > Manage > Data Protection > Replication. The replication type should be designated as "SRM Partner" for it to work with VMware SRM.

For detailed steps about configuring SRM and SRA, please refer to VMware SRM Installation and Configuration Guide and the IntelliFlash User Guide.

IntelliFlash VMware Integration

IntelliFlash systems have been interoperable with VMware ESXi for many years but the collaboration and integration only begins with certification. Several plugins and extensive VAAI support round out the relationship.

This section discusses all of the ways that the IntelliFlash system and VMware's product line are tightly integrated to give a better end user experience. The VMware Client Plugin and the VAAI integration is covered where VMware Site Recovery Manager is discussed in the Disaster Recovery section.

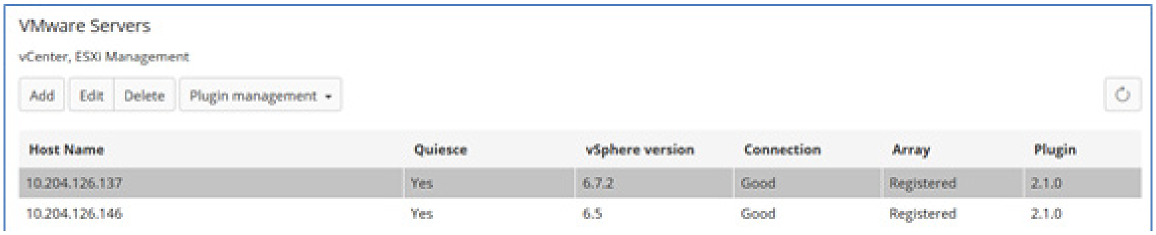
IntelliFlash VMware Client Plugin (VCP)

The vSphere Client Plug-In for IntelliFlash is a web-server based application. It extends the vSphere Client by adding specific IntelliFlash system functionality. Older versions of the plugin show the IntelliFlash icon the vSphere Client Home page Under Solutions and Applications. Here each icon launches the plugin for a single IntelliFlash system.

Newer versions of the plugin (v2.1 and higher) display the IntelliFlash icon labeled IntelliFlash Manager on the Home page of the vSphere Client or under the Menu pull down. Clicking the icon now stays within the vSphere client and all tasks are performed there. It is now very tightly integrated with the vSphere UI construct.

The plugin provides VMware data provisioning, cloning, recovery, and manageability, as well as the statistics of storage efficiency and performance.

In order for vSphere to identify the extended plugin, you must register it from the particular IntelliFlash it runs from. This is done from Settings > App-Aware > VMware Servers.



VMware Servers

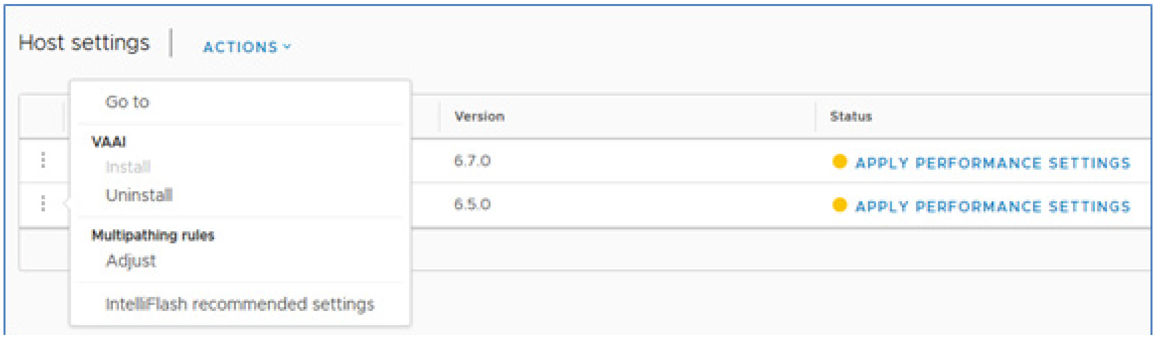
vCenter, ESXi Management

Host Name	Quiesce	vSphere version	Connection	Array	Plugin
10.204.126.137	Yes	6.7.2	Good	Registered	2.1.0
10.204.126.146	Yes	6.5	Good	Registered	2.1.0

Figure 24 – vCenter Servers Installed with the IntelliFlash Plug-In

ESXi Configuration with IntelliFlash VCP

To ensure no mistakes are made setting all of the ESXi host settings it's recommended to use the IntelliFlash VMware Client Plugin (VCP) 1. To access IntelliFlash VCP from vCenter, you need to add and register the vCenter Server in the IntelliFlash Web UI under Settings > App-Aware > VMware Servers.



Host settings

Version	Status
6.7.0	● APPLY PERFORMANCE SETTINGS
6.5.0	● APPLY PERFORMANCE SETTINGS

Figure 25 – IntelliFlash Performance Settings Thru VCP

For the ESXi hosts that have NFS access to the IntelliFlash system, use the IntelliFlash VCP on the vCenter Server, to apply the following settings. The NFS settings from Table 1 should also be changed using this method.

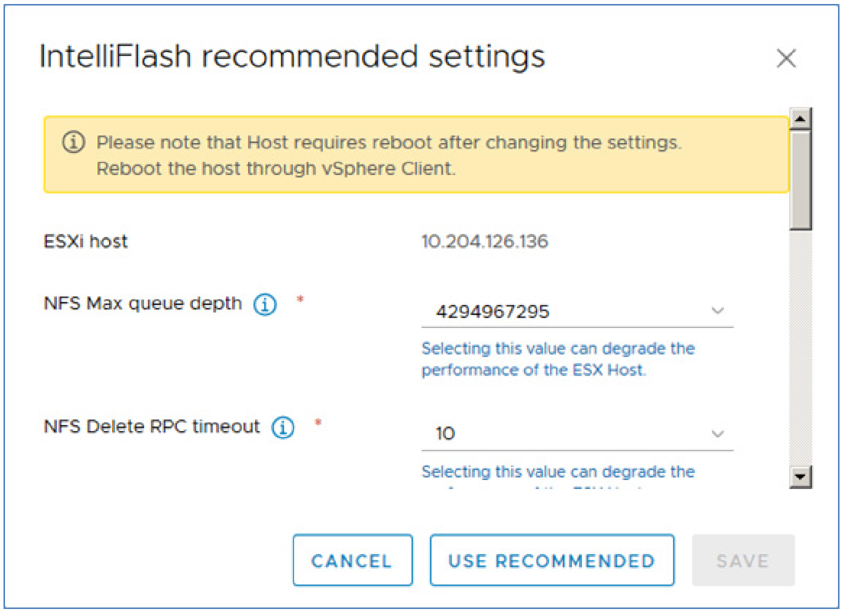
- Install IntelliFlash NAS-VAAI plug-in for VMware

For each of the ESXi hosts that have FC or iSCSI access to the IntelliFlash system use the IntelliFlash VCP on the vCenter Server to apply the following settings:

- Install the IntelliFlash multipathing (or SATP in VMware parlance) rules for both FC and iSCSI2.
- Set Maximum Disk I/O Size: 128
- Set ATS for VMFS Heartbeat: Disabled
- Set Disk.SchedNumReqOutstanding: 64 (CLI only)

Notes:

1. You need to reboot each of the ESXi hosts for the settings to take effect.
2. In ESXi 6.7 the SATP rules are pre-installed so this step is not necessary.


IntelliFlash recommended settings

Please note that Host requires reboot after changing the settings.
Reboot the host through vSphere Client.

ESXi host 10.204.126.136

NFS Max queue depth * 4294967295

Selecting this value can degrade the performance of the ESX Host.

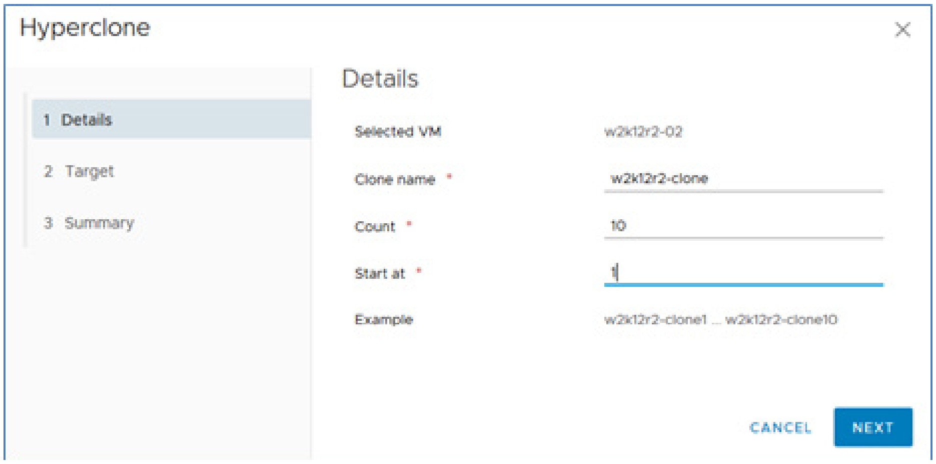
NFS Delete RPC timeout * 10

Selecting this value can degrade the

Figure 25 – IntelliFlash Performance Settings Thru VCP

Hyperclone

Typically, customers use vRealize Orchestrator to create workflows for multiple VM cloning. However, not everyone has vRealize. The HyperClone feature in VCP 2.x is a simple UI/template to allow users to clone multiple VMs directly in the vCenter UI.


Hyperclone

Details

Selected VM w2k12r2-02

Clone name * w2k12r2-clone

Count * 10

Start at * 1

Example w2k12r2-clone1 ... w2k12r2-clone10

Figure 27 – Hyperclone Wizard Using IntelliFlash vCenter Plugin

Hyperclone requires the IntelliFlash NFS VAAI plugin to be installed on the ESXi host. The plugin uses object-clone features to do file-based cloning within the same Pool. (See the section, ESXi Configuration with IntelliFlash VCP, on how to install the plugin.)

VCP 2.x will warn users if the NFS plugin is not installed when trying to perform a hyperclone on a VM. If the user continues without installing the NFS VAAI plugin normal cloning will be used.

The NFS VAAI fast clone is extremely fast. Most of the time will be on adding/registering those VMs to vCenter after the clone is done.

IntelliFlash and VAAI

VMware vSphere Storage APIs – System Integration or VAAI are a set of APIs or Primitives that enable communication between ESXi hosts and storage systems. The communication allows an ESXi host to tell the IntelliFlash system to do a task which offloads storage related operations to the storage system thereby reducing overhead on the ESXi server. This also helps improve performance of offloaded storage related operations significantly.

The full suite of IntelliFlash supported Primitives are listed in Table 6.

Area	Primitive	Description	Block	File
File System	Atomic Test & Set (Hardware Assisted Locking)	Enables granular locking of block storage devices, accelerating performance	Y	-
Clone	Full Copy (Xcopy, Extended Copy)	Commands the system to make a mirror of a LUN (Clone, VMotion)	Y	-
Thin Provision	Zeroing File Blocks	Communication mechanism for thin provisioning systems (Zero / Write Same)	Y	-
	Quota Exceeded Behavior	Allows vSphere to react before out-of-space condition (TP Soft Threshold)	Y	-
	UNMAP	Allow thin systems to clear unused VMFS space (Block Space Reclamation)	Y	-
	STUN	"Pause" a running VM when capacity is exhausted (Out of Space Condition)	Y	-
Clone	Full File Clone	Like Full Copy (Hardware Snap)	-	Y
	Fast File Clone	Clones at the file level - Offload linked clones	-	Y
Status	Extended Stats	vSphere can query file size, etc	-	Y

Table 6 – IntelliFlash Support VAAI Primitives

Cloning Primitives

IntelliFlash systems support fast VM cloning through VAAI for both block and file protocols. When cloning a VM on a datastore to the same system you can benefit from IntelliFlash VAAI support which offloads the VM copy operation from the server and the network and minimizes the VM clone time.

Block Cloning

On the block side there is Full Copy which requests that the IntelliFlash system perform a full copy of blocks on behalf of ESXi. It's primarily used in cloning and migration operations such as Storage vMotion.

A good practice is to reclaim space after a Storage vMotion which can be achieved through the VAAI primitive UNMAP.

File Cloning

On the file side there are several types.

- Full file clone – This is similar to Full Copy or XCOPY for block. It enables virtual disks or VMs to be cloned but not with Storage vMotion. The VM has to be powered off but this is still a useful primitive for deploying clones from a template.
- Fast file clone – Creates VM snapshots all offloaded to the IntelliFlash system. The VMs are linked clones created by the system rather than by the ESXi host using CPU and memory resources for the same task.

Thin Provisioning Primitives

There are several primitives under the thin provisioning umbrella like Write Same (Zero) which lets the storage system zero out requested blocks but the most interesting is around space reclamation. The rest of this section discusses this primitive for both block and file.

IntelliFlash systems support space reclamation on both SAN and NAS protocols. When a VM or VMDK is deleted and removed from a datastore, or when a VM is migrated to another datastore, the space that was originally occupied by the VM is freed up on the datastore. Depending on the protocol, the space may or may not be reclaimed instantaneously on the underlying storage.

The space that has been deleted or moved is referred to as dead space and the feature that can be used to reclaim space is called Space Reclamation which uses the SCSI command called UNMAP. This command when issued notifies the system that certain blocks are no longer needed by the ESXi host and can be reclaimed.

Dead space can be created in two places:

- VMFS – When a VM is moved (storage vMotion) or when a virtual disk is deleted, the space that was used to store the VM or virtual disk is now available again but the system does not know about it
- In-Guest – When a file has been moved or deleted on a virtual disk (.vmdk) inside of a VM, for example, from a guest filesystem, neither the underlying VMFS nor the storage system know about the block being freed by the virtual disk

Space reclamation in VMware equips the user to deal with both the types of dead space listed above. Space reclamation with VMFS varies depending on the version of ESXi. VMFS UNMAP has been supported in various forms since it was first introduced in ESXi 5.0.

File Based Space Reclamation

For NAS this is really straight forward and hands free. The NFS protocol by its nature implements this feature so the IntelliFlash system automatically reclaims the space when a file or a directory is deleted on an NFS share.

Block Space Reclamation Using UNMAP

In block-based storage implementations, the file system is managed by the host and not the storage system. Due to which, the storage system does not typically know when a file has been moved or deleted from a storage device and therefore does not know when to release the space.

For block protocols the IntelliFlash system supports VAAI Thin Provisioning UNMAP based on T10 standard SCSI commands. To reclaim the free space, the host needs to initiate the process by sending SCSI commands to request the storage to reclaim the space at the specific LBA range.

VMFS UNMAP is a manual process in vSphere 5.5 and 6.0 usually executed on demand as and when needed. In vSphere 6.5, UNMAP is an automatic process executed by the ESXi server as needed without any manual intervention.

VMFS UNMAP in vSphere 5.5 through 6.0

After VM deletion or Storage vMotion, the space is not automatically reclaimed on the storage device. To reclaim the space, you must issue ESXi shell commands manually.

To reclaim space on ESXi 5.5 and 6.0, use the following command:

```
esxcli storage vmfs unmap --volume-label=volume_label | --volume-uuid=volume_uuid --reclaim-unit=number
```

The number of VMFS blocks to UNMAP per iteration is an optional argument. If it's not specified a default value of 200 is used. For example:

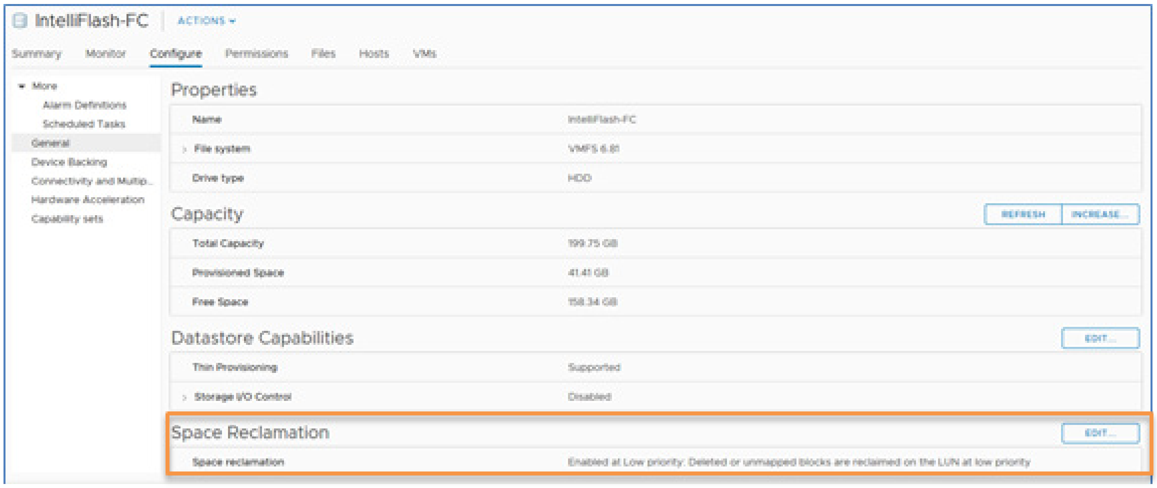
```
esxcli storage vmfs unmap -l DS_Tegile
```

ESXi 5.5 does not report how much space has been reclaimed. The space reclamation can be verified through the IntelliFlash Web UI where the original data size should be reduced accordingly on the LUN and Pool.

VMFS UNMAP in vSphere 6.5

In ESXi 6.5, VMware introduced automatic UNMAP support for VMFS volumes. This is applicable on the new VMFS version 6. An automated UNMAP crawler is introduced in ESXi 6.5 to work in tandem with VMFS-6 datastores that will run continuously in the background.

The vSphere UI can be used to see if Space reclamation is running against a particular datastore as shown below:



Properties

Name: IntelliFlash-FC
File system: VMFS 6.81
Drive type: HDD

Capacity

Total Capacity: 199.75 GB
Provisioned Space: 41.41 GB
Free Space: 158.34 GB

Datastore Capabilities

Thin Provisioning: Supported
Storage I/O Control: Disabled

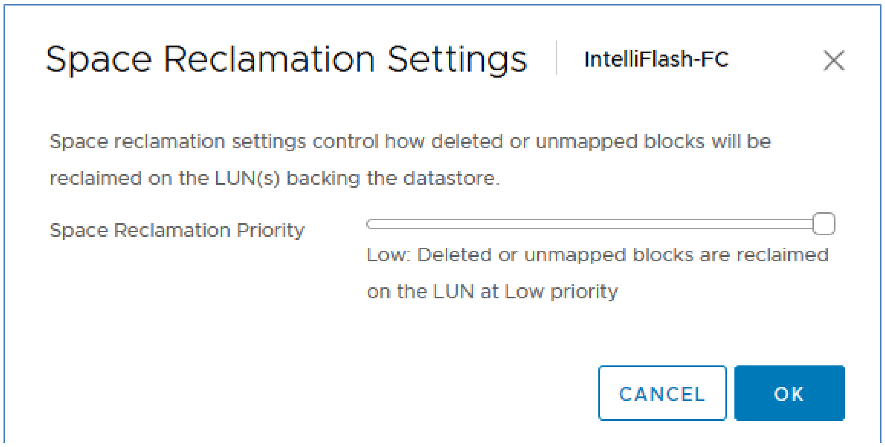
Space Reclamation

Space reclamation: Enabled at Low priority: Deleted or unmapped blocks are reclaimed on the LUN at low priority

Figure 28 – Space Reclamation on a Block Datastore

Currently in ESXi 6.5, there are two settings available via the vSphere UI (Figure 29) for setting the reclamation priority – None and Low. By default, the setting is Low.

Note: Automatic UNMAP in ESXi 6.5 is an asynchronous operation and reclamation is not instantaneous. It will take approximately 12 hours. If the user would like to reclaim the space immediately, the esxcli UNMAP operation can be run as described above. This is true for VMFS-5 datastores in ESXi 6.5 as well.



Space reclamation settings control how deleted or unmapped blocks will be reclaimed on the LUN(s) backing the datastore.

Space Reclamation Priority

Low: Deleted or unmapped blocks are reclaimed on the LUN at Low priority

Figure 29 – Space Reclamation Settings

In-Guest Space reclamation

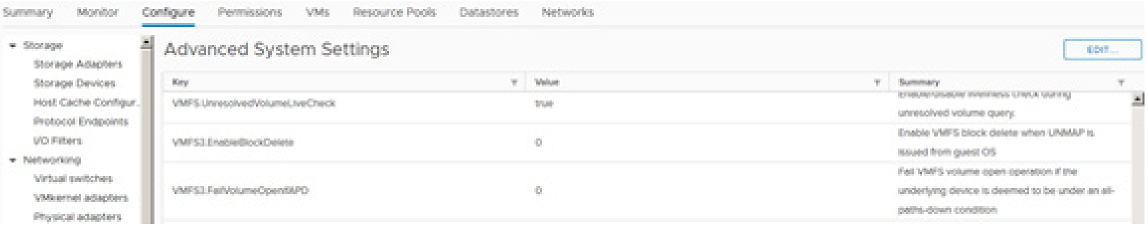
ESXi supports the UNMAP command directly from within a guest operating system allowing the OS to reclaim storage space. The process to free up space starts when a VM deletes files on a thin provisioned virtual disk. The guest OS then notifies the VMFS that space has been freed up by sending an UNMAP command where space is freed within the VMFS datastore. The UNMAP command is then passed on to the IntelliFlash system so that it can complete the reclamation process and free the blocks of space.

Support for this capability varies between OS and VMFS versions.

VMFS6 – This version of VMFS generally supports automatic reclamation requests from a guest OS. For an OS that supports automatic unmaps the results will depend on the size of the reclaim request. 1MB or multiples of 1MB will be processed but space less than 1MB or not aligned to 1MB will not be processed.

VMFS5 – Typically unmap commands from an OS on a VMFS5 datastore do not get passed to the system. You must run a manual command previously mention in section VMFS UNMAP in vSphere 5.5 through 6.0. For a guest operation that does support automatic unmapping they must have the following:

- A thin-provisioned virtual disk.
- The guest OS must be able to identify the virtual disk as thin.
- VM hardware must be of version 11 (ESXi 6.0) or later.
- The advanced setting EnableBlockDelete must be set to 1 (Default is 0).
- In ESXi 6.0 CBT must be off. In ESXi 6.5 CBT can be on.



Advanced System Settings

Key	Value	Summary
VMFS.UnresolvedVolumeLiveCheck	true	Enable/disable liveness check during unresolved volume query.
VMFS3.EnableBlockDelete	0	Enable VMFS block delete when UNMAP is issued from guest OS
VMFS3.FailVolumeOpenIfAPD	0	Fail VMFS volume open operation if the underlying device is deemed to be under an all-paths-down condition

Figure 30 – EnableBlockDelete Setting

Linux and Windows support both automatic and manual methods of UNMAP. Windows NTFS supports automatic UNMAP starting from ESXi 6.0. You can verify with the following CLI command on the windows command line utility:

```
c:\Users\Administrator.domain\fsutil behavior query DisableDeleteNotify
DisableDeleteNotify = 0
```

If DisableDeleteNotify is set to 0, UNMAP is ENABLED. Windows also supports manual UNMAP. If UNMAP is disabled in Windows Disk Optimizer can be used to reclaim space on-demand.

Linux file systems, unlike Windows, do not support automatic UNMAP by default. This behavior needs to be enabled during the mount operation of the file system. This is achieved by mounting the file system using the discard option.

```
sudo mount /dev/sdc /mnt/linuxunmaptest –o discard
```

The manual method of UNMAP with Linux file systems can be done by using

- fstrim command
- sg_unmap command

For Linux operating systems, in-guest UNMAP support is available starting from ESXi 6.5 and later & VM Hardware version 13 and later.

Performance Tuning

Optimal performance can be achieved by considering each of the following general best practices. These best practices are end to end from the IntelliFlash storage system through the storage or Ethernet fabric all the way to the guest OS running on an ESXi host:

- Bandwidth
    - Ensure the connectivity sufficient to deliver the expected throughput
    - Jumbo Frames
        - Enable Jumbo Frames end-to-end
        - Enable Jumbo Frames on in-guest virtual network adapter
    - Choose the VMXNET 3 virtual network adapter
- Load Balance
    - Evenly distribute the I/O across
        - The available physical paths between the hosts to the storage system
        - The ESX hosts
        - The IntelliFlash LUNs/shares
        - The storage controllers (Active/Active configuration)
        - IntelliFlash systems, if multiple are used
    - Enable Storage I/O Control on the ESXi host, if available
- Block Alignment
    - LUN/share block size to the application I/O request size
    - Guest OS partition offset to the LUN/share block size
    - Guest OS file system allocation unit size to the LUN/share block size
- Queue Depth
    - Configure HBA queue depth and ESX host LUN/NFS queue depth to maximize the number of concurrent I/Os to the IntelliFlash system without exceeding the system's max queue depth.
    - If running I/O intensive applications, such as SQL Server or Exchange on VMs, use VMware Paravirtual storage adapter, instead of the default LSI Logic SAS adapter. You should also increase the disk queue depth on the Paravirtual SCSI adapter on the guest OS to achieve the optimal I/O concurrency.

Understanding Queue Depth End-to-End

One of the keys to achieving optimal performance is to maximize the I/O concurrency between the hosts and storage without saturating any component in between. Once the physical environment has been built to maximize bandwidth and is properly load balanced the main focus of performance tuning is around queue depth.

The rule of thumb is to set the queue depth of the Guest OS and Host queue depth as big as possible, so long as the combined queue depth on all hosts do not exceed the queue depth supported by the storage system.

In order to fine-tune the queue depths for optimal performance, let's take a deep dive into the queue depths of various layers in between the VM and the storage system:

- Guest OS disk queue depth
- Device queue depth on ESXi.
- Disk.SchedNumReqOustanding (FC, iSCSI)
- NFS.MaxQueueDepth (NFS)
- HBA queue depth on the hypervisor (FC or iSCSI only)
- Device Queue Depth on IntelliFlash system

Guest OS Disk Queue Depth

The queue depth of the guest OS disk is determined by the virtual SCSI adapter that was used to configure the VM. LSI SAS adapter and VMware Paravirtual (PVSCSI) adapter are the two most commonly used virtual SCSI adapters.

LSI SAS adapter has a fixed queue depth whereas the PVSCSI adapter is user configurable and has a much higher maximum. See Virtual SCSI Controller Type section for queue depth defaults and maximums.

If the SCSI controller type is not Paravirtual on an existing VM, it can be re-configured to Paravirtual by editing the VM settings when the VM is in powered-off state. Before you make such changes, please check on the Paravirtual SCSI adapter support matrix at:
Configuring disks to use VMware Paravirtual SCSI (PVSCSI) controllers (1010398)

Configure PVSCSI Queue Depth

Here are the examples of how to increase the disk queue depth to 256 per device and to 1024 for the adapter:

Windows VM:
From the command line of your Windows VM, run this command and then reboot the guest OS:

```
REG ADD HKLM\SYSTEM\CurrentControlSet\services\pvscsi\Parameters\Device /v DriverParameter /t REG_SZ /d
"RequestRingPages=32,MaxQueueDepth=256"
```

One request ring page can hold 32 queued commands. With 32 request ring pages, it can support 1024 queued commands on the adapter. To verify the successful creation of registry entry:

1. Open the registry editor with the REGEDIT command
2. Browse to

```
HKLM\SYSTEM\CurrentControlSet\services\pvscsi\Parameters\Device.
```

3. Verify that the DriverParameter key exists with a value of RequestRingPages=32, MaxQueueDepth=254.

Linux VM:
Create a file of any name in the /etc/modprobe.d/ directory with the following line:

```
options vmw_pvscsi cmd_per_lun=254 ring_pages=32
```

Alternatively, append the following to kernel boot arguments (for example, on Red Hat Enterprise Linux edit /etc/grub.conf or on Ubuntu edit /boot/grub/grub.cfg):

```
vmw_pvscsi.cmd_per_lun=254
vmw_pvscsi.ring_pages=32
```

These queue depth changes require the guest OS reboot to take effect. After reboot, verify the changes through the following command line:

```
# cat /sys/module/vmw_pvscsi/parameters/cmd_per_lun
# cat /sys/module/vmw_pvscsi/parameters/ring_pages
```

Queue Depth on ESXi

Storage Device Queue Depth

On ESXi hypervisor, the disk device queue depth for FC and iSCSI LUNs is configurable on a per LUN basis through Disk.SchedNumReqOutstanding. It sets the maximum number of active I/O commands to a LUN allowed at any given time by the VMKernel.

The default value of Disk.SchedNumReqOutstanding is 32.

This setting is only applicable when there are multiple VMs using the same storage device on the host. Disk.SchedNumReqOutstanding controls the combined number of concurrent commands from all VMs to a disk device.

If there is only one VM using the LUN, this setting does not apply, HBA queue depth controls the maximum number of concurrent commands that the host can send to the LUN.

It's recommended that you set the ESXi disk device queue depth to 64 as a general practice. Since it is configured on a per LUN basis, every time a new LUN is created, it needs to be configured on the new LUN.

You can view the current setting with the following command:

```
esxcli storage core device list
```

You can change the setting with the following command:

```
esxcli storage core device set -d naa.xxx -O 64
```

NFS/Share Queue Depth

For NFS, the VMware advanced setting NFS.MaxQueueDepth determines the max number of concurrent I/O commands that can be sent to an NFS share at any given time.

By default, NFS.MaxQueueDepth is 4294967295, which is essentially no limit. This has been known to cause performance and connectivity issues with NAS storage devices from various vendors. NFS.MaxQueueDepth must be lowered to a reasonable value, such as 64.

It's recommended to use IntelliFlash vCenter Plug-in to configure NFS.MaxQueueDepth to 64 as a general practice.

ESXi HBA Queue Depth

This is applicable for FC or iSCSI. On ESXi, HBAs support two queue depth settings:

- Queue depth per device
- Queue depth per adapter.

Depending on the HBA vendor, model, driver and the ESXi version, the default values vary for these queue depth settings.

Check HBA Adapter Queue Depth on ESX/ESXi

1. Run the esxtop command in the service console of ESXi.
2. Press d.
3. Press f and select Queue Stats.
4. The value listed under AQLEN is the queue depth of the storage adapter.

The following is an example of ESX top with adapter status on an ESXi 6.5 host:

ADAPTR	PATH	NPTH	AQLEN	CMDs/s	READS/s	WRITES/s	MBREADs/s	MBWRTN/s
vmhba0	-	2	1024	0.00	0.00	0.00	0.00	0.00
vmhba1	-	2	2176	3.74	0.00	0.00	0.00	0.00
vmhba2	-	2	2176	0.00	0.00	0.00	0.00	0.00
vmhba64	-	6	1024	0.00	0.00	0.00	0.00	0.00

Table 7 – ESXtop Showing HBA Queue Depth (AQLEN)

In Table 7 vmhba1 and vmhba2 are FC ports a queue depth per adapter of 2176 as shown in the AQLEN column. The software iSCSI adapter, vmhba64, has the queue depth per adapter of 1024.

Configuring HBA Device Queue Depth

To check for a QLogic® driver run the following CLI commands on the ESXi host:

```
# esxcli system module list | grep qln
```

With the above result you can display the various configurable parameters for the HBA with the following command. In this scenario the QLogic module retrieved was qlnativefc.

```
# esxcli system module parameters list -m qlnativefc
```

The maximum queue depth setting varies but continuing with the QLogic example you can set the maximum queue depth value with:

```
# esxcli system module parameters set -p qlfxmaxqdepth=64 -m qlnativefc
```

The QLogic HBA max queue depth per device was set to 64. Reboot the ESXi host for the change to take effect. After the host boots up run the previous command that listed module's parameters

For information about configuring FC HBA queue depth on ESXi host on Cisco UCS, please refer to Tintri IntelliFlash UCS Configuration Guide. For more information about configuring FC HBA queue depth on ESXi, please refer to the VMware article at:
Changing the queue depth for QLogic, Emulex, and Brocade HBAs (1267)

Configuring the Software iSCSI Adapter Device Queue Depth

Use the following CLI command to set the maximum queue depth per device for software iSCSI adapter:

```
#esxcli system module parameters set -m iscsi_vmk -p
iscsivmk_LunQDepth=value
```

The iscsivmk_LunQDepth parameter sets the maximum number of outstanding commands, or queue depth, for each LUN accessed through the software iSCSI adapter. The default value is 128. Reboot your system for the change to take effect.

IntelliFlash System Queue Depth

On IntelliFlash systems, there are two layers of queue depth

- At the per device level – 64 per LUN/share and non-configurable
- At the interconnect level

For FC, the queue depth per port is 2048. For a controller, which has two FC ports, the queue depth per controller would be 4096. For an active/active configuration, the total queue would be 8192.

Protocol	Queue Depth	Configurable
FC	2048/port	No
iSCSI	2048/port	No
NFS	1024/controller	Yes[1]

Table 8 - IntelliFlash System Interconnect Queue Depth

Notes:

1. Please contact a Tintri support representative if you are considering changing the setting.

Queue Depth Tuning General Guidelines

To achieve the optimal performance, it's important to allow all the hosts to maximize the I/O queue to the storage system without overwhelming the storage device I/O command queue. Apply the following guidelines when tuning queue depth in an ESXi environment.

- Total QD of all hosts does not exceed the total QD of the IntelliFlash system
    - When controller fails in an active/active configuration, the total queue depth on the system would reduce to half.
    - The ESXi QD per LUN should be equal to or less than the FC or iSCSI HBA QD per LUN
- Total queue depth of LUNs on ESXi should not exceed the maximum QD on the HBA:
    - VMware software iSCSI, the number of iSCSI adapters is one per host.
    - The number of FC adapters is equivalent to the number of FC ports on the host
- The QD per LUN on the guest OS should not exceed the ESXi QD per LUN
- On a guest OS, the total QD of the LUNs on a virtual SCSI adapter should not exceed the QD per adapter.

Notes:

- If I/O is running on multiple I/O protocols simultaneously, the queue depth can be calculated independently on each I/O protocol.
- These guidelines assume that all LUNs or shares are shared by all hosts, and I/O load is balanced across all LUNs/shares, available paths, and storage system controllers

Example 1: Queue Depth Configuration with 8 hosts

The following is an example of an 8 host ESXi cluster sharing 8 NFS shares in an active/active configuration on IntelliFlash system:

- Total NFS queue depth on the IntelliFlash system is 2048 (2 x 1024)
- The NFS.MaxQueueDepth is set to 32 on each of the ESXi hosts
- The total queue depth on all hosts can reach maximum of 2048 = 8 hosts x 8 shares x 32 QD per share

This cluster has a total queue depth equal to the total NFS queue depth on the IntelliFlash system. Therefore, this is a good configuration.

If the NFS.MaxQueueDepth is set to 64, it runs the risk of exceeding the total queue depth of the system. When the storage system reaches queue full on NFS, it will reject the upcoming NFS requests, and the host will retry the requests and again the requests will be rejected.

In this way, it generates unnecessary network traffic as well as unnecessary loads on the storage system, as a result, it will slow down the performance and increase the I/O latency.

Example 2: Queue Depth Configuration

The following is an example of an 8 host ESXi cluster sharing 16 FC LUNs on IntelliFlash system in an active/active configuration, with 2 FC links per controller. On each of the hosts, there is a dual-port FC 8Gb QLogic HBA:

- Total FC queue depth on the system is 8192 = 2 ports x 2 controllers x 2048 QD per port
- The Disk.SchedNumReqOustanding is set to 64 per device on each ESX host. It is the same as the queue depth per device on the QLogic FC HBA. Therefore, it's a valid configuration.
- The total queue depth on all hosts in the cluster can reach a maximum of 8192 = 8 host x 16 LUNs x 64 QD per LUN

This is a good configuration, because the combined queue depth of all the ESXi hosts in the cluster does not exceed the total queue depth of the storage system.

- The total queue depth per host is 1024 = 16 LUNs x 64 QD per LUN

This is much lower than 4352 (2 ports x 2176 QD per port), the total queue depth that QLogic HBA would support. This is also valid.

If there were more than 16 FC LUNs shared across all the 8 hosts, and the Disk.SchedNumReqOutstanding remained at 64, it runs the risk of the combined queue depth on all hosts exceeding the total queue depth that the system can support.

When the system reaches a "queue full" condition, the system will reject the upcoming FC requests and return a "Queue Full" error status to the request. The host may retry. This may generate unnecessary load, which slows down the I/O and increase the I/O latency.

End-to-End Queue Depth Recommendation Summary

The following table lists the queue depth information of all layers between the ESXi host and the IntelliFlash system. In the sections below, instructions are given about how to modify these queue depth settings.

	Component / Setting	Default Queue Depth	Maximum Queue Depth	Recommended Queue Depth	Reboot
Windows Guest OS	LSI Logic SAS virtual Adapter	64 per device 128 per adapter	N/A	N/A	N/A
	VMware Paravirutal adatper	64 per device 256 per adapter	256 per device 1024 per adapter	Use VMware default	VM
Linux Guest OS	LSI Logic SAS virtual Adapter	32 per device 128 per adapter	N/A	N/A	N/A
	VMware Paravirutal adatper	32 per device 256 per adapter	256 per device 1024 per adapter	VM Tools VMware VSS HW Provider	VM
ESX	Disk.SchedNum-ReqOutstanding	32 per LUN	256 per LUN	64 per LUN	No
	NFS.MaxQueueDepth	4294967295 per share	4294967295 per share	64 per share	No
ESX Storage adapters	QLogic FC HBA	64 per device 2176 per FC port	256 per device N/A per port	Use default	ESXi
	VMware software iSCSI Adapter	128 per device 1024 per adapter	255 per device N/A per adapter	Use default	ESXi
IntelliFlash System	FC	2048 per port	N/A	N/A	N/A
	iSCSI	2048 per port	N/A	N/A	N/A
	NFS	1024 per controller	N/A	N/A	N/A

Table 9 - Queue Depth Summary

Additional Tuning Considerations

Round Robin Path Selection Policy Option IOPS

When configuring Round Robin Path Selection Policy, the IOPS option is set to a default of 1000. This means that the VMware SCSI module will issue 1000 commands through one path before it switches to another path.

VMware generally recommends leaving this option to the default value. This is because when the environment scales with greater number of VMs and datastores, VMware considers that the default setting associated with the Round Robin path policy to be sufficient.

In field deployments, some IntelliFlash customers have experienced performance improvement by setting IOPS=0 in their specific environments.

Therefore, it's recommended that you start with the VMware default. If you do not get the desired performance you may try setting IOPS=0.

To configure the IOPS option, use the following command on a per LUN basis:

```
# esxcli storage nmp psp roundrobin deviceconfig set --type=iops --iops=0 --device=<naa.xxx>
```

And then use the following command to check if the option has been configured properly:

```
# esxcli storage nmp psp roundrobin deviceconfig get -d naa.xxx
```

Check the output of the above command, which should contain following information:

```
IOOperation Limit: 1000
```

This setting change takes effect immediately without requiring host reboot.

Troubleshooting

Testing Jumbo Frames

Once jumbo frames is configured on the server, switch, and storage, it's necessary to test it to make sure it's configured properly end to end. To list the vmknic interfaces on the host, and identify the vmknic interfaces name for the data ports:

```
# esxcfg-vmknic -l
```

To test jumbo frames, do the following in the ESXi command shell:

```
# vmkping -d -s 8500 -I <vmknic interface> <IntelliFlash system floating IP address>
```

The option –d is used to make sure the host can ping the system by sending a jumbo message without breaking it into multiple packets. The packet size is exaggeratedly low to allow for frame headers and the way different switches may interpret the packet size. Trying to be exact can cause the ping to fail.

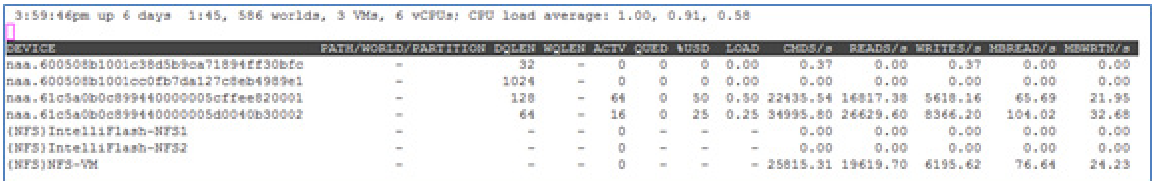
Execute the vmkping command on each vmknic interface on the host, and on each IntelliFlash system floating IP address on the data network. Sometimes it may require the ESX to reboot for the jumbo frame configuration change to work properly on the host.

ESXTOP

Esxtop is a powerful utility from VMware, which provides an interactive interface to provide I/O metrics over various devices attached to a VMware ESXi host. It's available through ESX service console or through ESXi shell.

Figure 31 shows an example of esxtop and how it's used to check the device queue depth as well as I/O metrics on the those devices. Simply run the following ESXi commands

1. Run the esxtop command in an ESXi shell
2. Press u.
3. Press f and select Queue Stats.
4. The value listed under DQLEN is the queue depth of the storage device. This is the maximum number of ESXi VMKernel active commands that the device is configured to support.



```
 3:59:46pm up 6 days  1:45, 586 worlds, 3 VMs, 6 vCPUs: CPU load average: 1.00, 0.91, 0.58

DEVICE                               PATH/WORLD/PARTITION DQLEN WQLEN ACTV QUED %USD  LOAD   CMDS/s  READS/s WRITES/s MBREAD/s MBWRTN/s
naa.600508b1001c38d5b9ca71894ff30bfc           -             32     -    0    0    0  0.00     0.37     0.00     0.37     0.00     0.00
naa.600508b1001cc0fb7da127c8eb4989e1           -           1024     -    0    0    0  0.00     0.00     0.00     0.00     0.00     0.00
naa.61c5a0b0c899440000005cffee820001           -            128     -   64    0   50  0.50 22435.54 16817.38  5618.16    65.69    21.95
naa.61c5a0b0c8994400000005d0040b30002          -             64     -   16    0   25  0.25 34995.80 26629.60  8366.20   104.02    32.68
(NFS)IntelliFlash-NFS1                         -              -     -    0    -    -     -     0.00     0.00     0.00     0.00     0.00
(NFS)IntelliFlash-NFS2                         -              -     -    0    -    -     -     0.00     0.00     0.00     0.00     0.00
(NFS)NFS-VM                                    -              -     -    0    -    -     - 25815.31 19619.70  6195.62    76.64    24.23
```

Figure 31 – ESXtop Example Showing Device Queue Depth (DQLEN)

In the above example, there are four storage devices and 3 NFS shares with active I/O. Their queue depth per LUN varies from 32 to 1024 as shown in DQLEN column, with the active queue depth between 16 and 64, as shown in ACTV column.

On these devices, workload is running between 22k and 35k IOPS per device, as shown the CMD/s which totals the READS/s and WRITES/s columns.

When troubleshooting performance issues, the esxtop storage device I/O information can helpful to identify possible bottlenecks. For more information, please refer to VMware knowledge base at: Using esxtop to identify storage performance issues for ESX / ESXi (1008205)

VMware Logs

Please refer to the following VMware knowledge base article for the VMware logs and their locations on various vSphere versions:

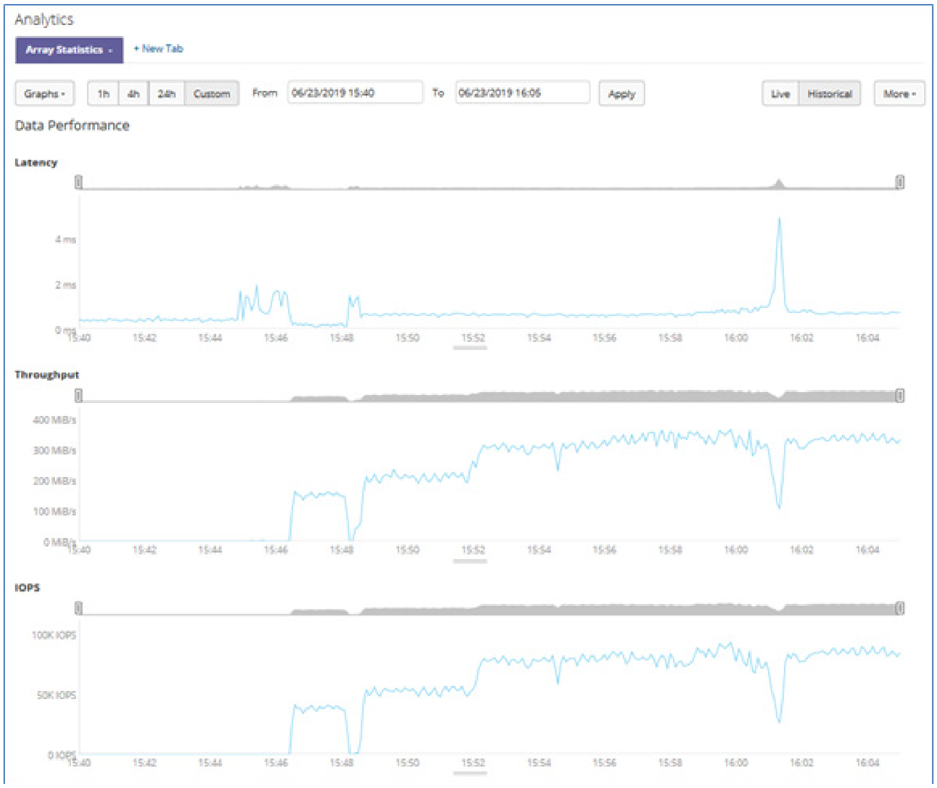
Location of log files for VMware products (1021806)

Among those VMware logs, the Core VMKernel logs on the ESXi host are the logs that are frequently referenced for troubleshooting storage related issues.

IntelliFlash Analytics

The IntelliFlash system offers extensive analytics through the embedded Web based GUI. It provides the information about the usage and activities about CPU, cache hit, FC, iSCSI, NFS, SMB, disk statistics etc. Such information is powerful for troubleshooting.

It keeps real time analytics and historical analytics. Figure 32 shows an example of the IntelliFlash historical analytics screen:



From 06/23/2019 15:40 To 06/23/2019 16:05

Figure 32 – IntelliFlash Historical Analytics

The historical analytics information can be exported in graphical or .csv format for further data analysis for a specified period of time. Historical analytics helps you analyze issues that occurred during the recent past, or the I/O load over a period of time. It's very useful in troubleshooting and performance tuning.

Disclaimers

This document contains recommendations for building a generic VMware vSphere implementation. It does not take into account requirements for security, performance, resilience, and other operational aspects that may vary for individual customer deployments. If recommendations in this document conflict with current operational guidelines, those existing policies should be given higher priority. Tintri provides this document as is without warranty of any kind, either express or implied, including the implied warranties of merchantability of fitness for a particular purpose.

Tintri, Tintri logo, Tegile, IntelliFlash and IntelliCare are registered trademarks or trademarks of Tintri Corporation or its affiliates in the U.S. and/or other countries. Citrix and XenServer are trademarks for Citrix Systems, Inc. and/or one or more of its subsidiaries, and may be registered in the United States Patent and Trademark Office and in other countries. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries. The NVMe word mark is a trademark of NVM Express, Inc. QLogic is a registered trademark of QLogic Corporation. Oracle is a registered trademark of Oracle and/or its affiliates. VMware, VMware vSphere, VMware vCenter, and VMware ESXi are registered trademarks or trademarks of VMware, Inc. in the United States and/or other jurisdictions. All other marks are properties of their respective owners. References in this publication to Tintri products, programs, or services do not imply that they are intended to be made available in all countries.