QoS for Cisco HyperFlex in Cisco ACI Deployments

Contents

Introduction

You can enable quality-of-service (QoS) features in a data center fabric to provide differentiated services to the different types of traffic traversing the fabric. QoS is necessary when there is congestion in the fabric to ensure that traffic gets the service that it needs. A fabric can become oversubscribed if it is not properly sized to accommodate the load on the network. Sizing should factor in not only steady-state usage, but also peak loads and failure scenarios. Failures can cause traffic to take an alternate path and increase the load on specific nodes and links in the fabric, resulting in congestion. Design choices can also cause congestion in the fabric – for example, at transition points where the bandwidth goes from high-speed (40-/100-Gbps) links to lower-speed (25-/10-/1-Gbps) links. These types of bandwidth transitions typically occur in the access layer where it connects to endpoints or other networks. For example, a 40-/100-Gbps ACI fabric may use 10-/25-/40-Gbps links to connect to a Cisco Unified Computing System™ (Cisco UCS®) domain where the endpoints reside or to connect to external networks such as an Inter-Pod network (IPN) in a Cisco® ACI Multi-Pod fabric.

Cisco HyperFlex™ clusters have stringent latency, drop, and bandwidth requirements that must be maintained at all times. To provide critical storage and data services and maintain a healthy cluster during periods of congestion, you should deploy QoS policies end-to-end to provide the QoS that HyperFlex needs. QoS policies are always in place in the Cisco UCS fabric; they are deployed by the HyperFlex Installer during the initial install of the cluster. However, if the HyperFlex cluster connects to a larger data center fabric where there is a risk of congestion, you should extend QoS into the data center fabric as well. For example, if the data center fabric is a Cisco Application Centric Infrastructure (ACI) fabric, you should enable QoS policies in the ACI fabric. If the HyperFlex™ cluster is a stretched cluster that connects to an ACI Multi-Pod fabric, you should enable QoS policies in the ACI fabric and extend it across the IPN as well. The QoS policies in the Cisco UCS and ACI fabrics should also be aligned so that HyperFlex traffic can receive consistent QoS end-to-end.

To provide QoS, a comprehensive set of QoS features and capabilities are available on Cisco platforms such as HyperFlex servers with Cisco Virtual Interface Cards (VIC) and Cisco UCS Fabric Interconnects in the Cisco UCS fabric, and the Cisco Nexus 9000® Series switches in the ACI fabric.

This paper focusses on the QoS design for a HyperFlex cluster connected to an ACI fabric. The QoS design is based on the HyperFlex with ACI design in the Cisco HyperFlex 3.5 stretched cluster with ACI 4.0 MultiPod fabric solution. The design and deployments guides for this Cisco Validated Design (CVD) are provided in the References section of this document.

Requirements

The QoS design presented in this document is specifically for connecting HyperFlex (standard or stretched) clusters to a Cisco ACI fabric. The goal of this design is to ensure that HyperFlex traffic, primarily storage data and storage management traffic, receives the quality of service it requires when traversing an ACI fabric. The design requirements addressed in this document for providing QoS for HyperFlex in an ACI deployment are:

- Preserve a consistent end-to-end QoS policy for HyperFlex traffic
- Meet minimum bandwidth requirements for HyperFlex storage
- Incorporate and align HyperFlex QoS with other traffic in an ACI fabric for an integrated QoS policy

HyperFlex Requirements

HyperFlex requires low latency, high bandwidth network connectivity between nodes in the cluster to:

- Maintain a healthy storage cluster and provide storage services to applications hosted on the cluster. HyperFlex delivers performance and high availability through parallel distribution and replication of storage data between nodes in the cluster. The system continuously optimizes the stored data through real-time, always-on, deduplication and compression. To maximize application performance, HyperFlex dynamically places and moves data between storage (memory, caching and capacity) tiers, maintained on nodes in the cluster. All of these storage functions rely on the network connectivity between nodes in the cluster.
- Meet application requirements, specifically read and write latencies for accessing storage provided by the HyperFlex cluster. Latency requirements for enterprise applications depend on the type of application, but they are generally in the range of tens of milliseconds. The latency experienced by an application typically includes processing by different software and hardware components in the application layer and outside the application, as well as forwarding and transport across multiple network hops. The latency budget for storage is, therefore, a fraction of the overall latency, typically in the ones of milliseconds range.

The network is critical for the storage services and overall health of any HyperFlex cluster, but it plays an even greater role in HyperFlex stretched clusters. A Hyper Flex stretched cluster provides high-availability for the virtualized server infrastructure in a data center by failing over to a second data center in the event of a failure. HyperFlex achieves high-availability by evenly distributing server nodes across both data centers. The data centers are typically in different locations, for example, different sites in a metropolitan area. A stretched cluster also requires a Witness node, deployed in a third location and with reachability to both data centers. Witness node serves as a tie-breaker that determines which site is primary in split-brain failure scenarios. The connectivity between Witness site and each data center should have at least 100 Mbps of bandwidth and round-trip time (RTT) of 100 milliseconds or less. For connectivity between data centers, a HyperFlex stretched cluster must have 10Gbps of bandwidth and RTT of 5 milliseconds or less between sites. For the most current and uptodate information on HyperFlex stretched clusters, please refer to the Operating Cisco HyperFlex HX Data Platform Stretch Clusters white paper provided in the References section of this document.

ACI Fabric Requirements

The ACI fabric QoS requirements for supporting HyperFlex clusters are summarized below:

- **At a minimum**, the ACI fabric QoS should preserve the class-of-service (CoS) values in any HyperFlex traffic originating from a Cisco UCS domain so that the same CoS values can be used for providing QoS in the domain receiving the traffic. If CoS is not preserved, the receiving domain can incorrectly classify and queue the HyperFlex storage traffic, resulting in poor storage performance. The receiving UCS domain can also drop the traffic if it is assigned to a class that does not support Jumbo Frames. By default, ACI supports Jumbo Frames but does not preserve CoS.
- **If there is a risk of congestion**, the ACI fabric should also have QoS policies in place to ensure that the HyperFlex platform receives its fair share of the fabric bandwidth and the minimum bandwidth required to maintain a healthy storage cluster. A HyperFlex cluster can forward traffic through the ACI fabric during its normal operation (for example, traffic between HyperFlex UCS domains in a stretched cluster) or temporarily during routine maintenance tasks (for example, firmware upgrades in the UCS fabric). It can also occur during failure scenarios such as a port or link failure on a server that connects to one side of the UCS fabric which can cause some of the HyperFlex traffic to traverse the ACI fabric. In all these scenarios, ACI fabric will be involved in the forwarding of HyperFlex traffic as it provides connectivity between Fabric Interconnects in a UCS domain or between UCS domains.
- Lastly, any QoS implemented in the ACI fabric for HyperFlex traffic has the potential to affect other traffic on the same fabric. You should therefore carefully evaluate your current ACI QoS design and adapt the QoS design discussed here to provide QoS for both HyperFlex and the other traffic traversing the fabric.

Deployment models for HyperFlex with ACI

The QoS design discussed in this document is applicable to the following deployment models:

- HyperFlex standard cluster connecting to an ACI fabric (Figure 1)
- HyperFlex stretched cluster connecting to a single-site ACI fabric (Figure 2)
- HyperFlex stretched cluster extended across an ACI Multi-Pod fabric (Figure 3)

Figure 1 shows a HyperFlex standard cluster attached to an ACI fabric. In this deployment model, the ACI fabric Leaf switches are involved in the forwarding of cluster traffic when the traffic between HyperFlex nodes crosses both Fabric Interconnects, as shown by dotted red and green arrows in Figure 1.

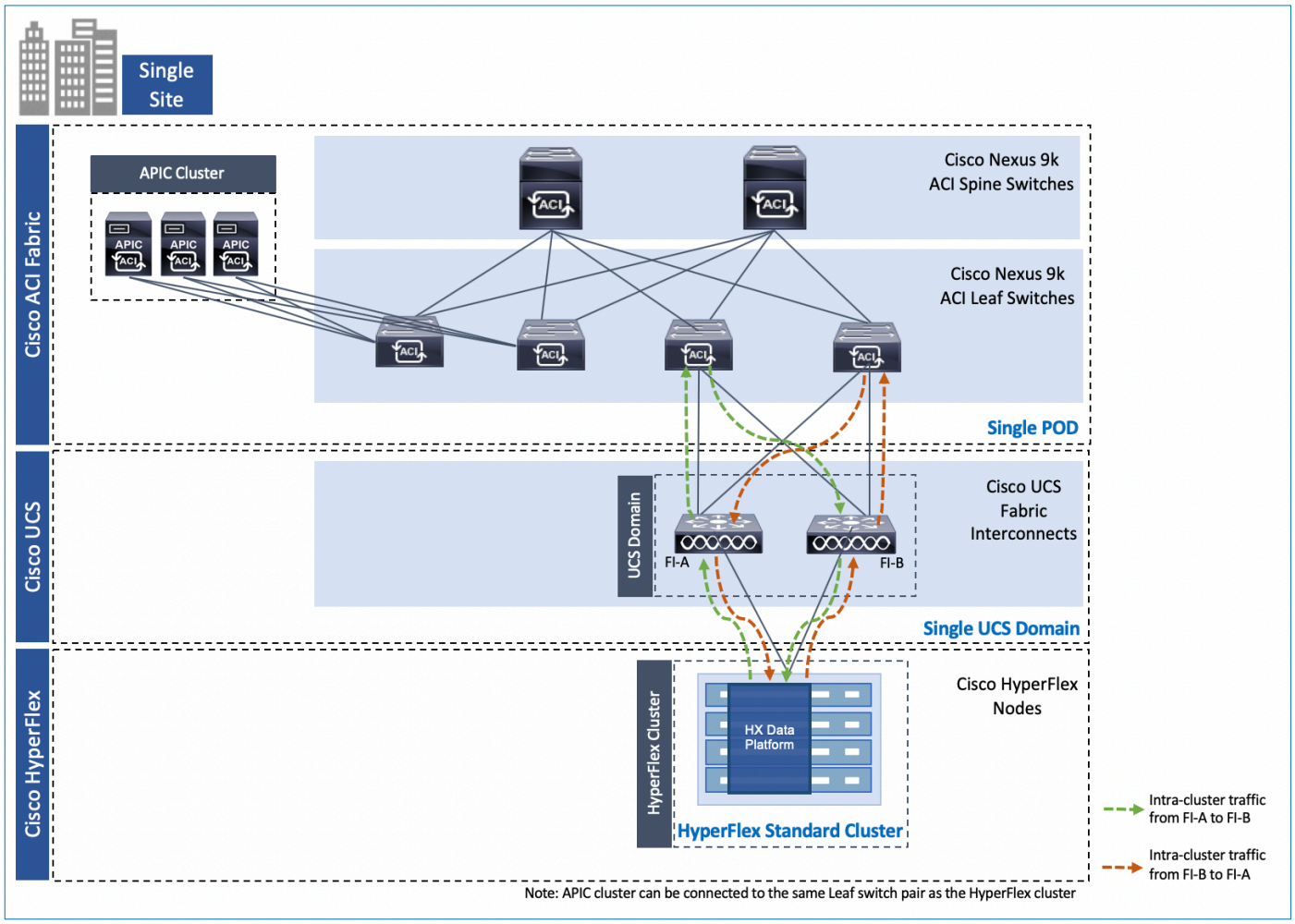

Figure 1.
HyperFlex standard cluster connected to a Cisco ACI fabric

Figure 2 shows a HyperFlex stretched cluster attached to a single-site ACI fabric. In this deployment model, the HyperFlex cluster traffic between the two Cisco UCS domains in the cluster are always through the ACI fabric, as shown by the solid orange arrows in the figure. ACI fabric also forwards the cluster traffic within a Cisco UCS domain when the traffic needs to be forwarded across both Fabric Interconnects, as shown by the dotted yellow and green arrows in the figure. The upstream leaf switches provides the forwarding in this case.

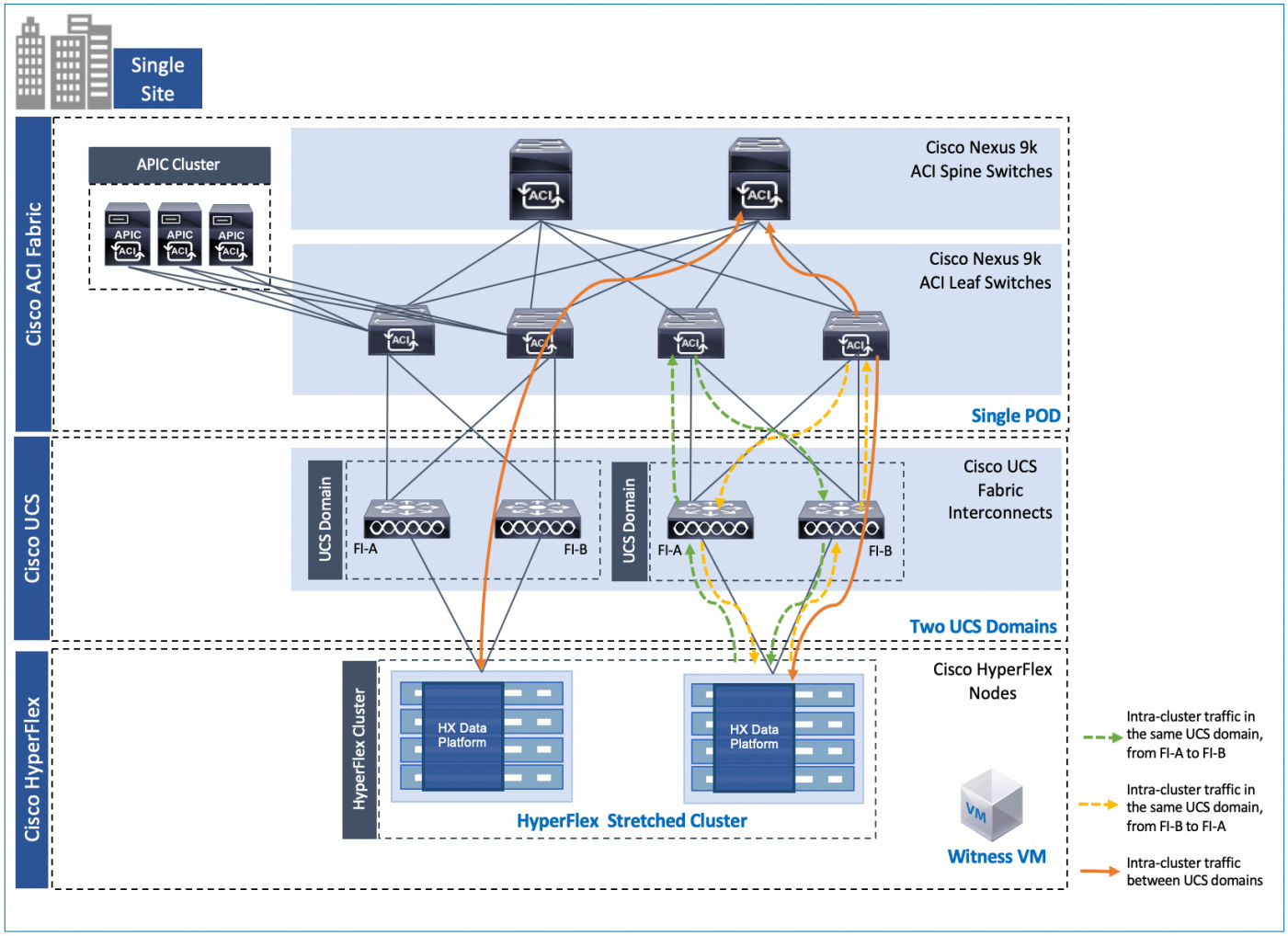

Figure 2.
HyperFlex stretched cluster connected to a single-site ACI fabric

Figure 3 shows a HyperFlex stretched cluster extended across an ACI Multi-Pod fabric. In this deployment model, the HyperFlex cluster traffic between the two Cisco UCS domains in the cluster are always through the ACI fabric and the IPN, as shown by the solid orange arrows in the figure. As in the previous two deployment models, ACI fabric also forwards the cluster traffic within a Cisco UCS domain when the traffic needs to be forwarded across both Fabric Interconnects, as shown by the dotted yellow and green arrows in the figure. The upstream leaf switches provides the forwarding in this case.

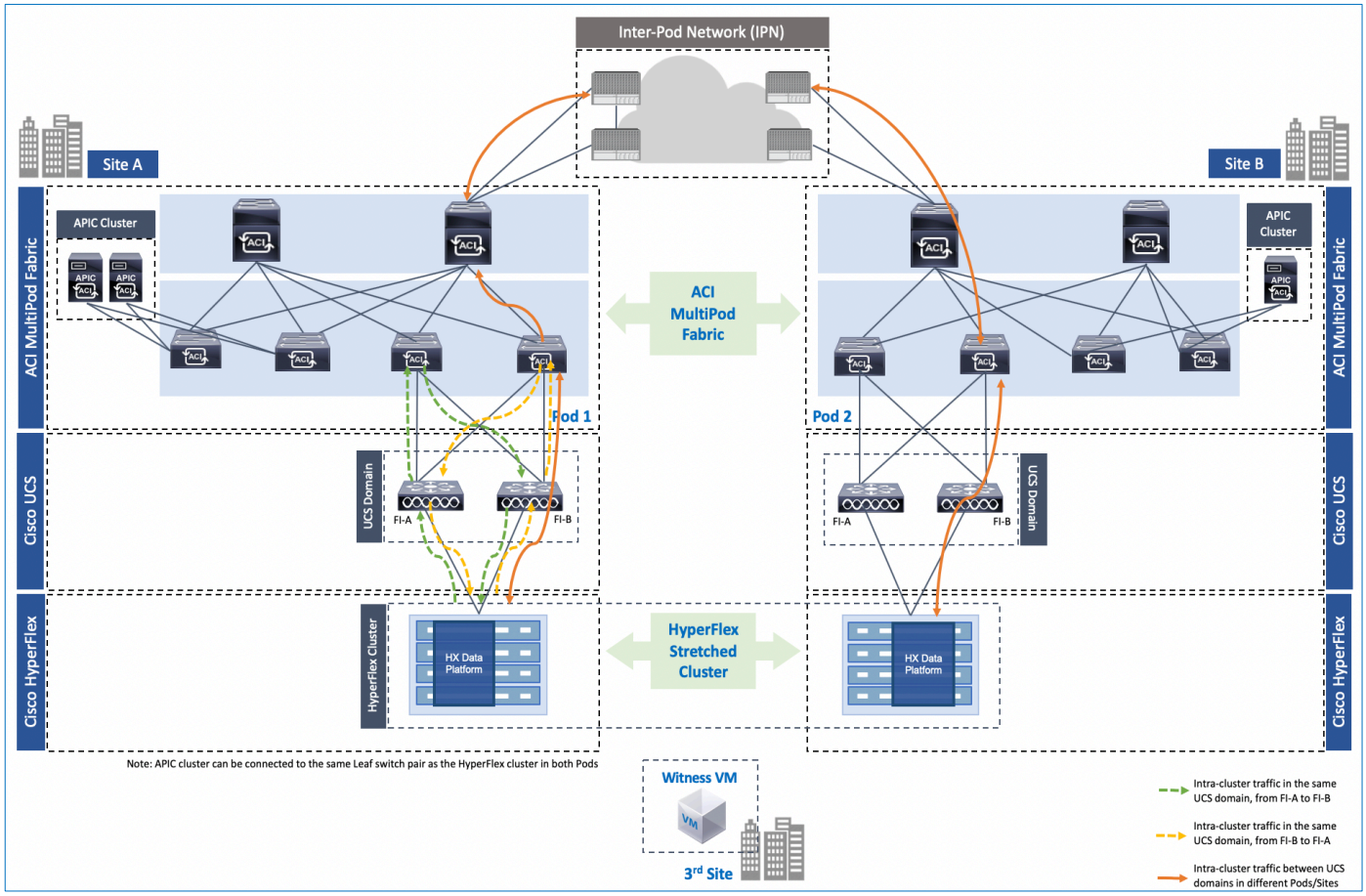

Figure 3.
HyperFlex stretched cluster connected to an ACI Multi-Pod fabric

Additional details and other requirements for a HyperFlex stretched cluster are provided in the Operating HyperFlex HXDP Stretch Clusters white paper (refer to the References section of this document).

Cisco UCS QoS design for HyperFlex traffic

The QoS design discussed in this section helps ensure that HyperFlex traffic receives the necessary QoS in the Cisco UCS fabric during periods of congestion. The QoS policies for these situations are also deployed by the HyperFlex installer.

The Cisco HyperFlex solution uses Cisco UCS technology to deliver a hyperconverged, high-performance, and highly available data center platform. A HyperFlex system consists of servers that connect to redundant network fabrics provided by a pair of Cisco UCS Fabric Interconnects. In a HyperFlex standard cluster, the servers are in a single UCS domain and each server is dual-homed to the two Fabric Interconnects that make up that domain. In a HyperFlex stretched cluster, the servers are distributed across two UCS domains and each server is dual-homed to the Fabric Interconnects in their domain.

Cisco HyperFlex servers are equipped with at least one Cisco Virtual Interface Card (VIC) to connect to the Cisco UCS fabric. Each server uses two ports to connect to the Fabric Interconnects (FI-A, FI-B) in the Cisco UCS domain (Figure 4). The Cisco VIC is a PCIe standards-compliant network interface card (NIC) that can support up to 256 virtual NICs (vNICs).

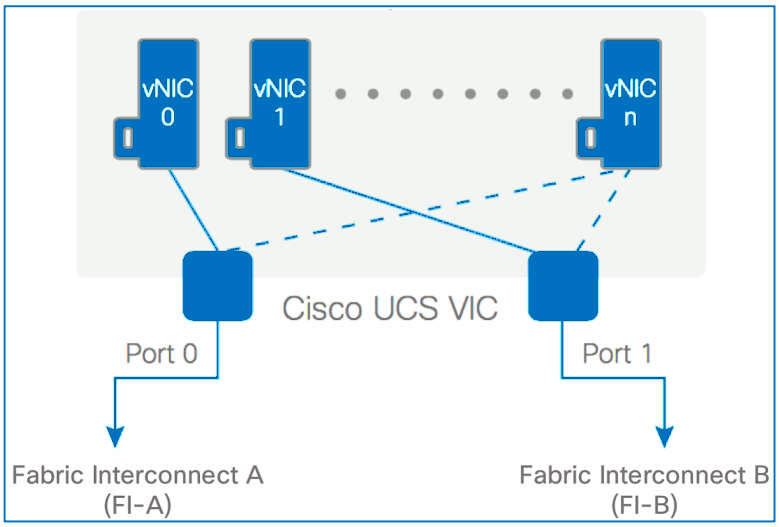

Figure 4.
Cisco HyperFlex server connectivity to UCS fabric

Each HyperFlex server is configured for eight vNICs, two for each HyperFlex traffic type, to provide redundant connectivity from the server to the network fabrics (FI-A, FI-B). Table 1 shows the different HyperFlex traffic types, the vNICs for each traffic type, and the UCS fabric (FI-A, FI-B) they connect to.

Table 1. Cisco HyperFlex server vNIC to Cisco UCS fabric mapping

HyperFlex traffic type	Virtual NIC (vNIC)	
	To Fabric Interconnect A (FI-A)	To Fabric Interconnect B (FI-B)
HyperFlex storage data	storage-data-a	storage-data-b
Virtual-machine network traffic	vm-network-a	vm-network-b
HyperFlex management	hv-mgmt-a	hv-mgmt-b
vMotion	hv-vmotion-a	hv-vmotion-b

Note that if replication between HyperFlex clusters is enabled for protecting virtual machines hosted on a cluster, HyperFlex will use the management vNICs for this traffic.

The vNICs for each HyperFlex traffic type connect to different UCS fabrics (FI-A, FI-B) as shown in Figure 5. For example, the two storage data vNICs (storage-data-a, storage-data-b) connect to Fabric Interconnect A and B, respectively, to provide redundant paths to the upstream network for HyperFlex storage traffic.

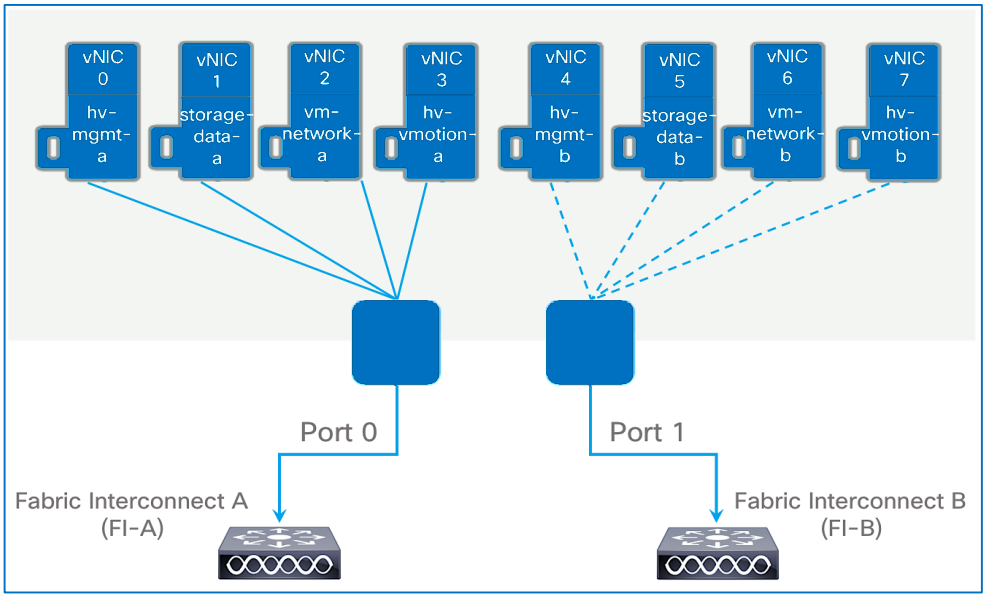

Figure 5.
Cisco HyperFlex server vNIC connectivity to Cisco UCS fabric

Each HyperFlex server is also setup at the virtualization layer to distribute the different HyperFlex traffic types across different UCS fabrics. For instance, under normal operation, each HyperFlex server will be configured at the hypervisor level to use vNIC 0 to FI-A as the primary interface for management traffic and vNIC 5 to FI-B as the primary interface for storage data traffic as shown in Figure 6. If there is a port, link, vNIC or FI level failure, traffic will failover to the backup vNIC for the corresponding traffic and forward traffic using the other UCS fabric. Figure 6 shows the primary and secondary paths for each traffic type from a HyperFlex server to the UCS fabrics. Figure 7 shows a logical view of the primary and secondary paths for a given HyperFlex server. Note that virtual machine networks uses both virtual NICs for forwarding traffic.

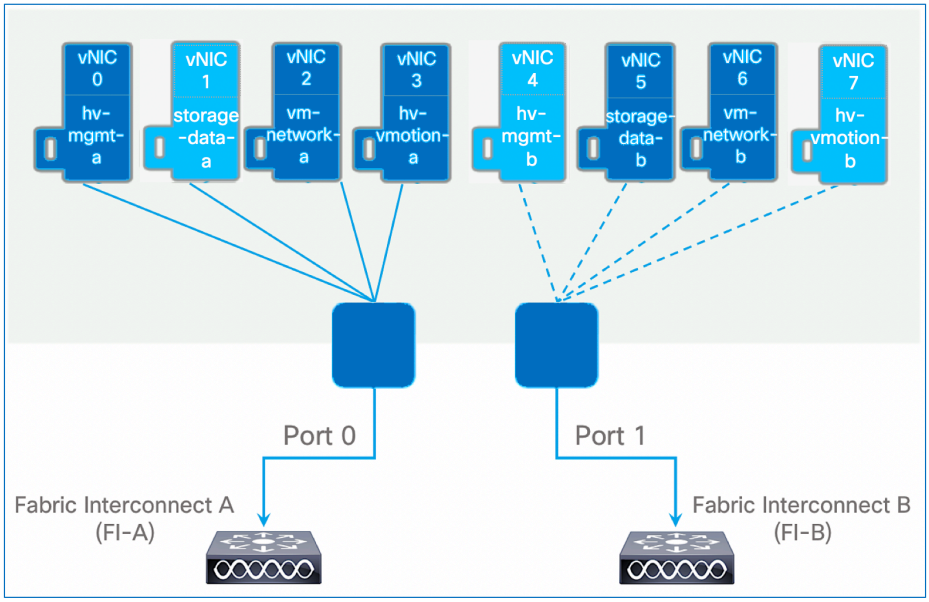

Figure 6.
Distribution of HyperFlex server traffic across UCS fabrics

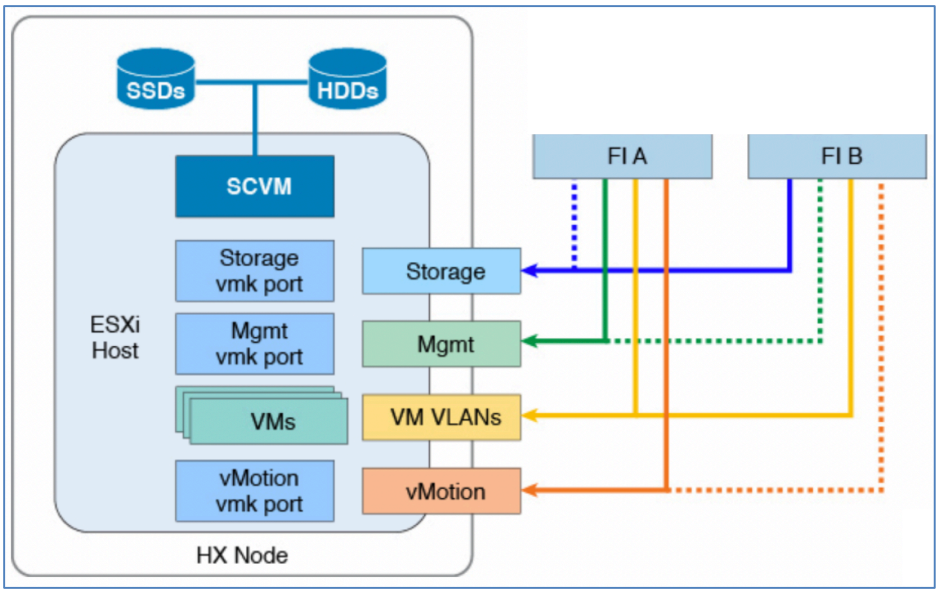

Figure 7.
HyperFlex server traffic to UCS fabrics—Logical view

Classification and marking of HyperFlex traffic

The Cisco VIC classifies and marks traffic originating from the HyperFlex servers by using QoS policies that the HyperFlex Installer deploys. The QoS policies mark each traffic type using a different CoS value. The policies are applied to the different HyperFlex vNICs using vNIC templates that create and configure the vNICs on each server. The vNIC templates are part of a larger Cisco UCS service profile template that configures each HyperFlex server.

Table 2 shows the **QoS Policy** applied to the vNICs for each HyperFlex traffic type. The **QoS Policy** specifies the CoS values used to mark the IEEE 802.1p header of the traffic received from the servers.

Table 2. QoS Policy applied to vNICs for each HyperFlex traffic type

HyperFlex traffic type	Virtual NIC (vNIC)		Cisco UCS QoS policy name	Class of service (CoS)
	To Fabric Interconnect A (FI-A)	To Fabric Interconnect B (FI-B)		
HyperFlex storage data	storage-data-a	storage-data-b	Platinum	5
Virtual-machine network traffic	vm-network-a	vm-network-b	Gold	4
HyperFlex management	hv-mgmt-a	hv-mgmt-b	Silver	2
vMotion	hv-vmotion-a	hv-vmotion-b	Bronze	1

Figure 8 shows the **Platinum** QoS policy used for HyperFlex storage traffic. The **Priority** field in the policy specifies the CoS value that you should use to mark the incoming traffic from the host. The **Host Control** field specifies whether to mark the traffic using the CoS specified by the **Priority** field in this policy or to trust the marking in the traffic received from the host. This field is set to **None** for all HyperFlex QoS policies to indicate that the UCS QoS policy will determine the CoS and any marking on the received traffic from the host will be re-marked using the **Priority** field in the policy. The QoS policy can also use the **Rate** field to rate-limit the bandwidth a given vNIC can use. Typically, the bandwidth available on the uplink port of a server is shared by the different vNICs that use the port. HyperFlex does not limit the bandwidth a given vNIC can use, thus allowing one HyperFlex vNIC to use the full bandwidth available if the other vNICs are not using it.

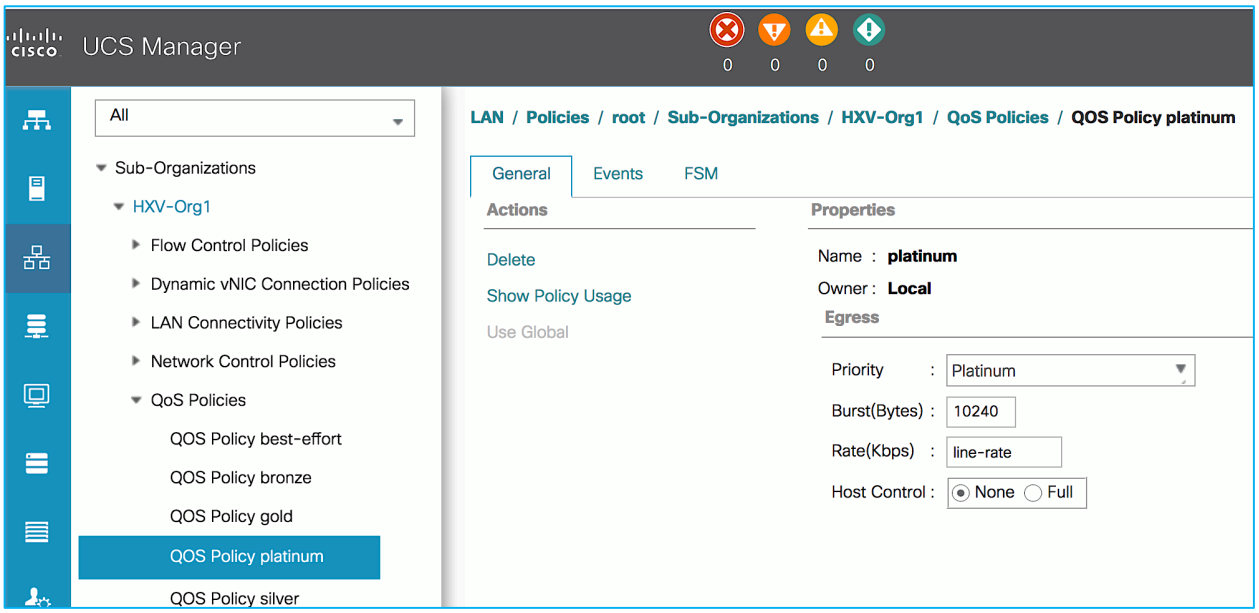

Figure 8.
Platinum QoS policy for HyperFlex storage traffic

As stated earlier, QoS policies are applied to the vNICs using vNIC templates that create and configure the vNICs on each server. A QoS policy (for example, platinum) is applied to a vNIC template (for example, **storage-data-a**) as shown in Figure 9.

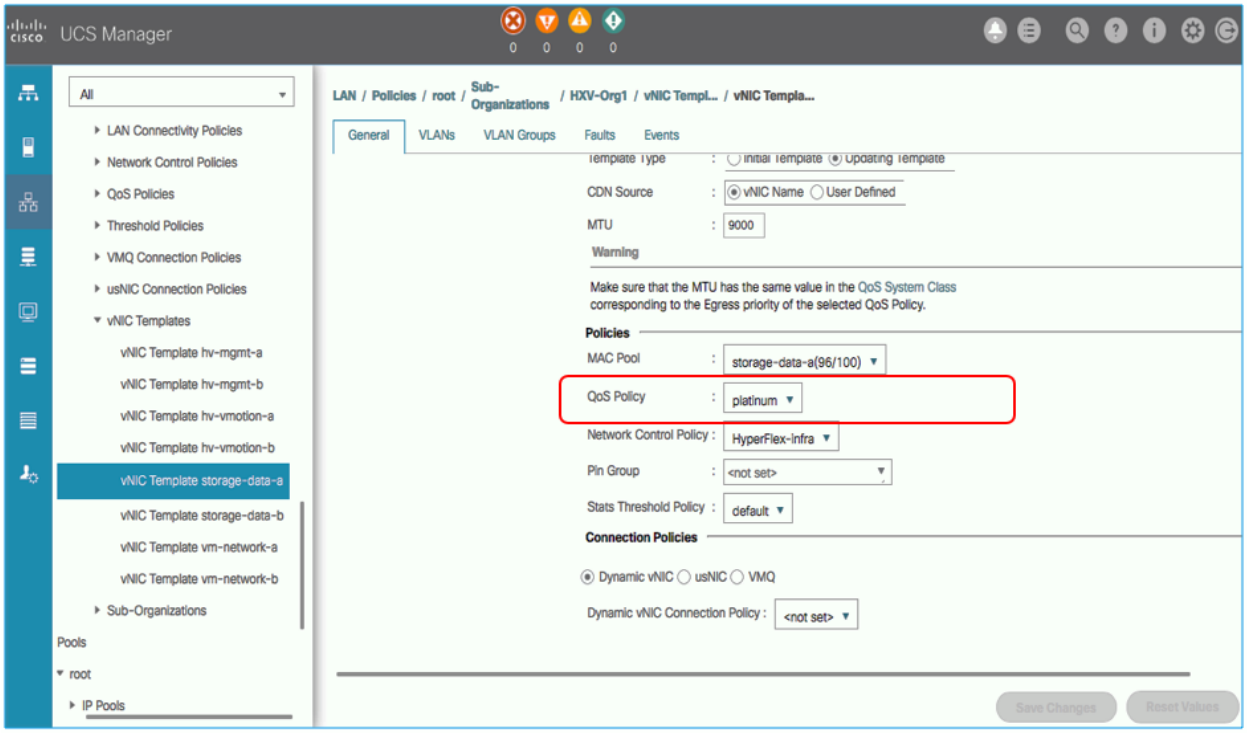

Figure 9.
QoS policy applied to server vNIC template

Table 3 shows the QoS policies and the vNIC templates the HyperFlex installer deploys.

Table 3. Server QoS policies and vNIC templates deployed by the HyperFlex installer

HyperFlex traffic type	vNIC template	QoS policy name
HyperFlex storage data	storage-data-a storage-data-b	Platinum
Virtual-machine network traffic	vm-network-a vm-network-b	Gold
Virtual-machine network traffic	hv-mgmt-a hv-mgmt-b	Silver
vMotion	hv-vmotion-a hv-vmotion-b	Bronze

As a result of the QoS policies that the HyperFlex installer deploys, any traffic sent from a HyperFlex server is marked with the CoS values necessary to provide QoS for that class of traffic in the upstream network fabrics.

Congestion management

The Cisco UCS fabric was designed to carry different types of traffic across a unified fabric using a Data Center Ethernet (DCE) network. To provide lossless transport for storage traffic across a shared DCE network, Priority Flow Control (PFC) was defined by IEEE 802.1Qbb. For storage traffic such as Fibre Channel traffic, PFC helps ensure zero packet loss during congestion by using pause functions to provide link-layer flow control. PFC also allows you to prioritize traffic on the same Ethernet link by using the CoS field in the IEEE 802.1p header of the Ethernet frame. In HyperFlex deployments, PFC is used for storage data traffic to ensure zero packet loss within the Cisco UCS domain(s).

The DCE network within the Cisco UCS fabric divides the bandwidth into eight virtual lanes or classes; two are reserved for the internal control and management of the system. The remaining six virtual lanes are available for traffic being switched through the fabric. The incoming traffic shares the available fabric bandwidth across these six virtual lanes. All traffic the UCS fabric receives is first classified based on its CoS value and then queued and scheduled using Weighted Round Robin (WRR) to meet the bandwidth allocation defined for that class. The bandwidth allocation specifies the minimum bandwidth guaranteed for that class during periods of congestion.

A global **QoS System Class** determines the bandwidth allocation and other parameters that determine the QoS that each class of traffic receives in the Cisco UCS domain. Figure 10 shows the **QoS System Class** that the HyperFlex installer deploys for a HyperFlex UCS domain. In a HyperFlex stretched cluster, the HyperFlex installer deploys the same parameters for the QoS system class in both UCS domains—and any post-install changes, which are performed manually, should be applied to both domains. The **Priority** field is the CoS value used to classify traffic into a given class. **Packet Drop** is allowed on all classes except for the **Platinum** class used by HyperFlex storage data. The **MTU** deployed by HyperFlex Installer is typically 9216B for the platinum (storage) and bronze (vmotion) classes and 1500B or **normal** MTU for other classes.

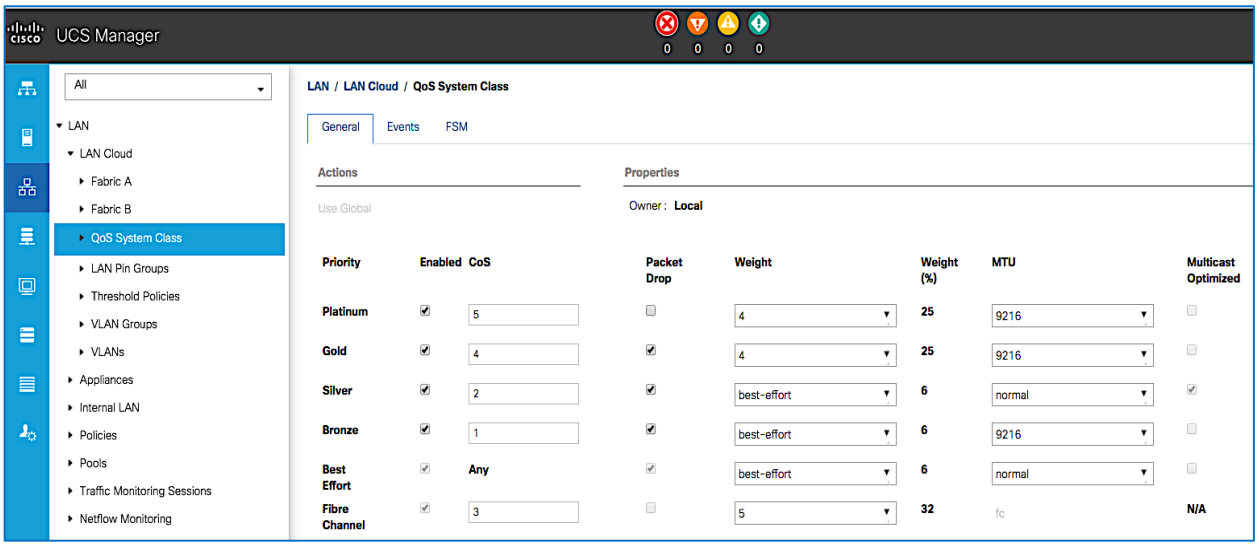

Figure 10.
QoS system class deployed by Cisco HyperFlex Installer

You can change the QoS system class parameters that the Installer deploys if necessary—for example, in Figure 10, the **Gold** class for HyperFlex virtual-machine network traffic has been changed to 9216B. No MTU changes are necessary on Cisco UCS 6454 Fabric Interconnect models because the default (**normal**) MTU is 9216B; MTU also cannot be changed on QoS classes that allow drops.

Note: Any changes to the global **QoS System Class** such as the MTU size, packet-drop policy, weight, etc. can cause a brief data-forwarding interruption as buffers and queues are adjusted to accommodate the changes. Some Fabric Interconnect models may also require a reboot for the changes to take effect. If a reboot is required, the system will indicate it on the UCSM GUI.

To provide QoS for HyperFlex traffic within the Cisco UCS fabric, the traffic needs to be classified and queued to one of the classes in the **QoS System Class**. The Cisco UCS platform classifies the traffic based on its CoS value. For traffic originating from HyperFlex servers, the **QoS Policy** deployed by the HyperFlex Installer marks the traffic with the correct CoS values (based on the HyperFlex traffic type) and the Cisco UCS platform uses this CoS to classify the traffic. However, the Cisco UCS platform does not do CoS marking for traffic received (ingress) on its uplinks. Instead, it uses the CoS value in the received traffic to classify and provide QoS for that traffic. It is assumed that the traffic was marked correctly at the originating end or in the ACI fabric, well before it is received by the UCS domain on its uplinks. Therefore, you should correctly mark any traffic forwarded by the ACI fabric for it to be classified and queued correctly within the UCS fabric. The next section provides two methods for preserving the original CoS marking in any traffic forwarded by the ACI fabric. One of these methods is necessary in HyperFlex deployments in order to preserve the CoS for HyperFlex traffic received from other Cisco UCS domains connected to the same ACI fabric.

The complete CLI-based QoS configuration for the Cisco UCS fabric is provided in Appendix A.

Cisco ACI QoS design for HyperFlex traffic

To provide a differentiated service, ACI can classify, mark, and queue the incoming HyperFlex traffic based on the type of traffic received from HyperFlex clusters. ACI can classify the traffic based on the incoming VLAN tag which ACI maps to an Endpoint Group (EPG) or based on the VLAN tag (or EPG) and COS value. ACI receives various types of traffic from HyperFlex clusters as outlined below:

- Application traffic from virtual machines hosted on the cluster, destined to other application components or services reachable through the ACI fabric.
- Storage traffic from virtual machines and ESXi hosts to access data stored on HyperFlex datastores.
- Management traffic between HyperFlex storage controller virtual machines and server nodes in the cluster, including management traffic to/from ESXi hosts.
- Storage traffic for the distributed placement of stored data across nodes in the cluster, including data moved between storage tiers and storage replication traffic.
- VMware vMotion traffic when moving virtual machines between nodes in the same cluster or other clusters.

The intracluster traffic between HyperFlex nodes and controller virtual machines in a cluster is critical to the health of the cluster, and therefore it is the primary focus of this design.

ACI is not always involved in the forwarding of intracluster HyperFlex traffic. It depends on whether the HyperFlex cluster is a standard or stretched cluster. A HyperFlex standard cluster connects to Fabric Interconnects in a single Cisco UCS domain, and ACI is typically not involved in the forwarding. The HyperFlex installer configures each traffic type on each HyperFlex node in the cluster to send traffic on a primary interface that connects to one of the Fabric Interconnects in the UCS domain, resulting in all traffic of a given type to be locally switched through one Fabric Interconnect. However, a given HyperFlex traffic type can get switched across **both** Fabric Interconnects in certain failure scenarios. For example, if there is a failure in the primary in-band management virtual NIC (vNIC) on one node, the in-band management traffic between this node and other nodes in the cluster would be forwarded across both Fabric Interconnects. Routine maintenance activities such as a server reboot for a policy change or a Fabric Interconnect upgrade can also result in forwarding across both Fabric Interconnects. Since ACI provides connectivity between Fabric Interconnects in a UCS domain, ACI will forward the traffic between Fabric Interconnects, as shown in Figure 1. The red and green arrows represent HyperFlex traffic between nodes that use different Fabric Interconnects as its primary path for a given traffic type. In these scenarios, if there is a mismatch or lack of alignment between the UCS and ACI fabric QoS for HyperFlex, traffic can get black-holed and de-stabilize the cluster. Note that, in this case, the sending and receiving Cisco UCS domains are the same from an ACI perspective.

For a HyperFlex stretched cluster, ACI is always involved in the forwarding of intracluster HyperFlex traffic since ACI interconnects the two Cisco UCS domains in the cluster as shown in Figure 2 (single-site ACI fabric) and Figure 3 (ACI Multi-Pod fabric). For intracluster traffic between nodes in the same UCS domain, the forwarding behavior is same as a HyperFlex standard cluster where ACI is involved in the forwarding only when traffic traverses both Fabric Interconnects in the UCS domain. The dotted arrows represent intracluster traffic between nodes in the same UCS domain while the solid arrow represents traffic between UCS domains.

ACI QoS Levels

ACI uses system-level QoS classes or levels to provide QoS. When there is congestion, the QoS level that a given class of traffic is mapped to determines the QoS it receives within the ACI fabric. Six user-configurable QoS levels are available as of Cisco APIC Release 4.0(1); earlier releases supported three user-configurable QoS levels. Each QoS level represents the comprehensive QoS that traffic classified into that level receives, including the congestion policy (Tail Drop, Weighted Random Early Detection), scheduling policy (Strict Priority, Weighted Round Robin), queuing policy (control, limit), bandwidth allocation, and Fibre Channel over Ethernet (FCoE)-specific QoS (Priority Flow Control, no-drop CoS level). For a given QoS level, you can configure the parameters as shown in Figure 11.

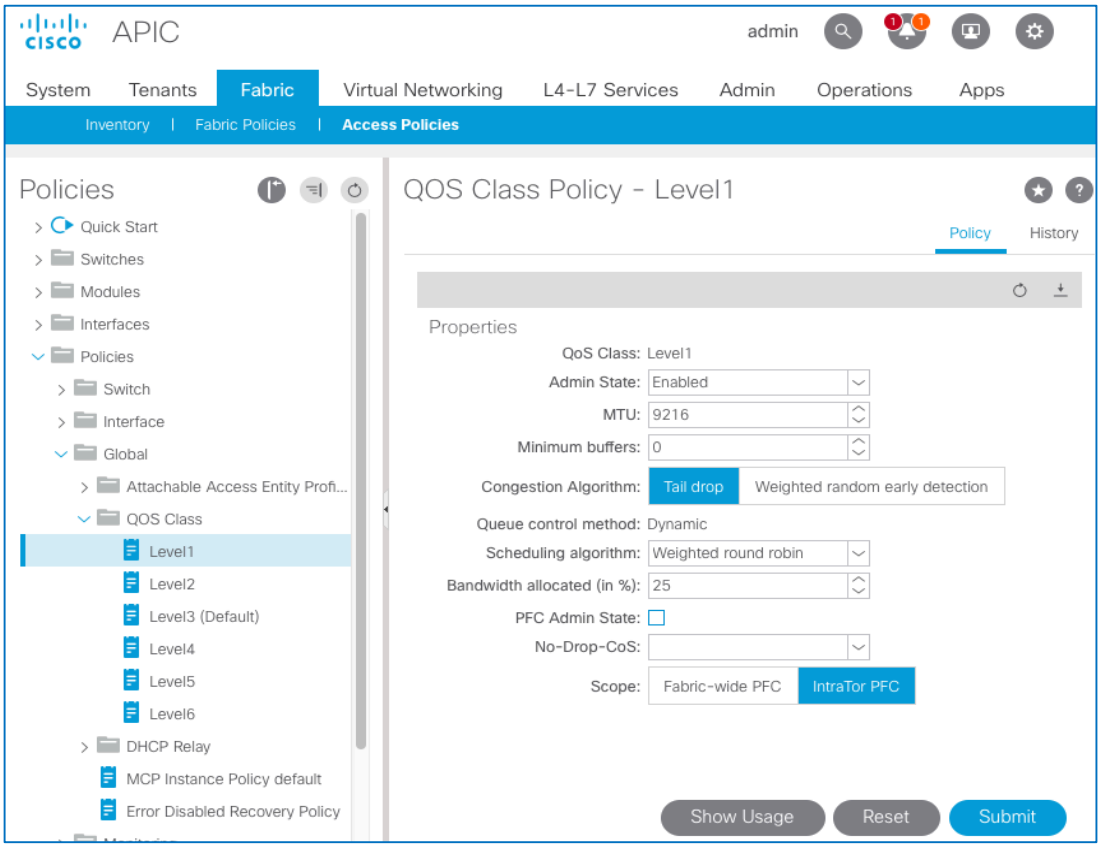

Figure 11.
ACI QoS Level parameters

ACI Qos Design

The QoS design described in this section ensures that HyperFlex traffic gets its fair share of the ACI fabric bandwidth during periods of congestion. At a high level, the design provides the following:

- Classifies incoming HyperFlex traffic to QoS classes or levels. ACI uses QoS levels to provide QoS within the fabric, including minimum bandwidth guarantees for each class of traffic.
- Preserves CoS for HyperFlex traffic as it is forwarded through the ACI fabric, enabling the receiving network or domain to provide QoS based on the original CoS marking in the traffic.
- Enables QoS for HyperFlex traffic in the IPN network when an ACI Multi-Pod fabric is used to interconnect two HyperFlex stretched cluster sites.

Classification and marking in fabric

As stated earlier, ACI classifies all incoming traffic to a QoS class or level. Classification typically occurs on the ingress ports of a leaf switch. The QoS level represents the comprehensive set of QoS polices that ACI applies to the traffic during congestion. By default, ACI classifies all incoming traffic to a default QoS class (**level 3**). However, you can also classify and mark the incoming traffic to other QoS classes by using one of the methods outlined below:

- Classify based on the source EPG and mark using the **QoS Class** option for the EPG
- Classify and mark using a **Custom QoS** policy at the source EPG level. Classification is based on either IP Differentiated Services Code Point (DSCP) or CoS value of the traffic received by that EPG.
- Classify and mark using the **QoS Class** option in an EPG contract

Source EPG and QoS Class policy

Though all three methods are valid, classifying HyperFlex traffic based on the source EPG is preferred as it provides the granularity and flexibility that most deployments will need. It is also simpler when QoS classification is based on the incoming traffic type since it is also what ACI uses to classify incoming traffic to an EPG. For more details on the ACI design for HyperFlex, see the Cisco HyperFlex 3.5 stretched cluster with ACI 4.0 MultiPod fabric CVD documents provided in the References section of this document.

Table 4 provides a summary of the HyperFlex traffic types and the EPGs that they map to in the above CVD. The traffic can be classified based on the source EPG (and traffic type) to the ACI QoS classes that determine the QoS the traffic will receive in the ACI fabric when there is congestion.

Table 4. ACI EPGs and QoS levels for HyperFlex traffic types

HyperFlex traffic type	HyperFlex QoS class	Source EPG	ACI QoS class
HyperFlex storage data	Platinum	HXV-CL1-StorData_EPG	Level 1
Virtual-machine network traffic	Gold	HXV-A-App_EPG, HXV-A-Web_EPG, ...	Level 2
HyperFlex management	Silver	HXV-IB-MGMT_EPG	Level 4
vMotion	Bronze	HXV-vMotion_EPG	Level 5

For a given EPG, you can classify and map the traffic to a QoS level by using the QoS class parameter, as shown in Figure 12.

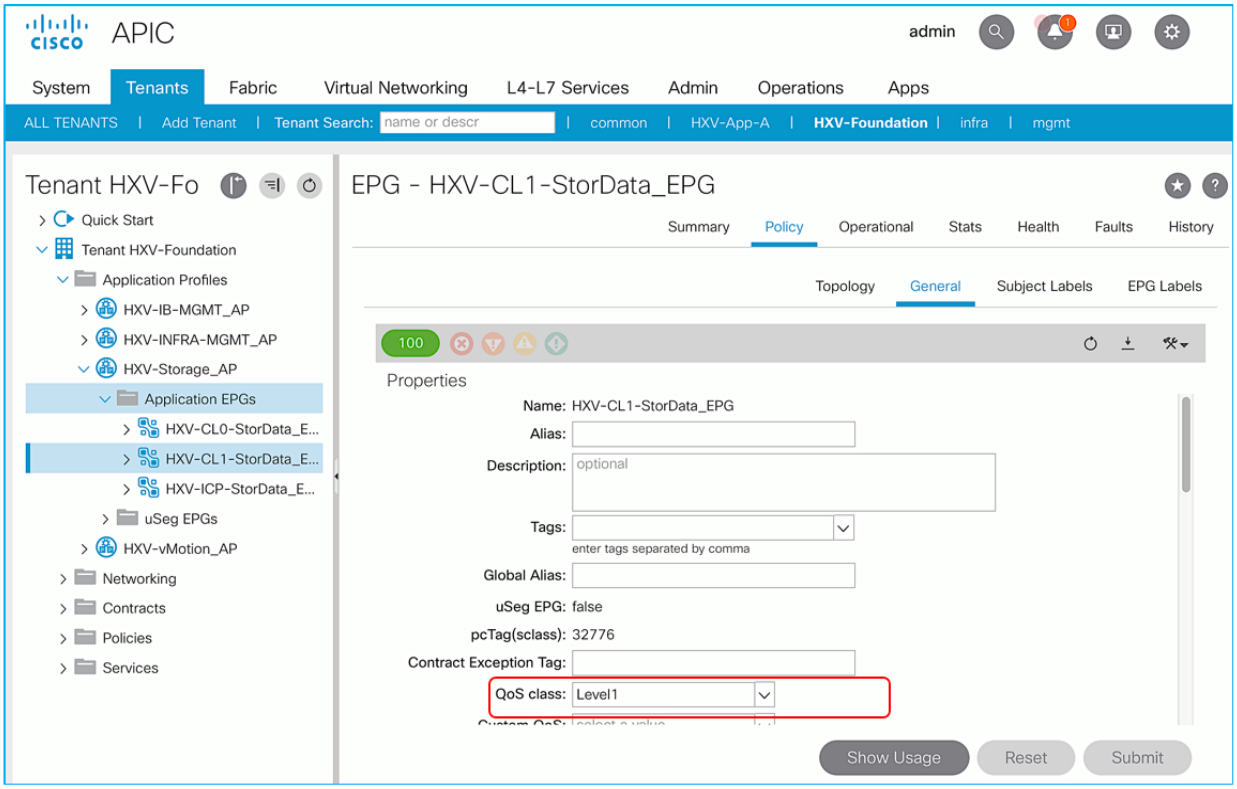

Figure 12.
EPG to QoS Level mapping

Source EPG and Custom QoS policy (optional)

You also can use a **Custom QoS** policy to classify incoming traffic. Custom QoS policies provide more granularity than the **Source EPG option** discussed in the previous section. Instead of classifying all incoming traffic in the EPG to a QoS level, you can base the classification on the CoS or DSCP value in the incoming traffic for a given EPG. Custom QoS policies also allow you to re-mark CoS and/or DSCP for traffic forwarded across the ACI fabric, but re-marking is not necessary in HyperFlex deployments. However, a **Custom QoS policy** has the following requirements:

- If you use a **Custom QoS** policy, you must enable the **Dot1p Preserve** feature. This feature is discussed in the next section.
- **Custom QoS** policy is not supported if you use a **COS to DSCP Translation** policy. ACI Multi-Pod deployments **may** require a CoS-to-DSCP translation policy. If so, use other methods to classify traffic.
- If you also use a QoS policy in the EPG contract, the EPG contract policy takes precedence.

Based on these guidelines, any single-site ACI fabric deployments can use **Custom QoS** policies to classify incoming HyperFlex traffic. However, if it is an ACI Multi-Pod deployment, you cannot use a **Custom QoS** policy if you also want to use a **COS to DSCP Translation** policy to ensure the QoS in the IPN and remote Pod. The COS to DSCP Translation policy is discussed in greater detail later in this document.

Congestion management

Each HyperFlex traffic type receives QoS based on the system-level QoS class or **level** it is mapped to and the parameters for that class. The parameters include congestion policy, scheduling policy, queueing parameters such as bandwidth allocation percent, and flow control. Figure 13 shows the ACI QoS levels used in this design and the associated QoS parameters.

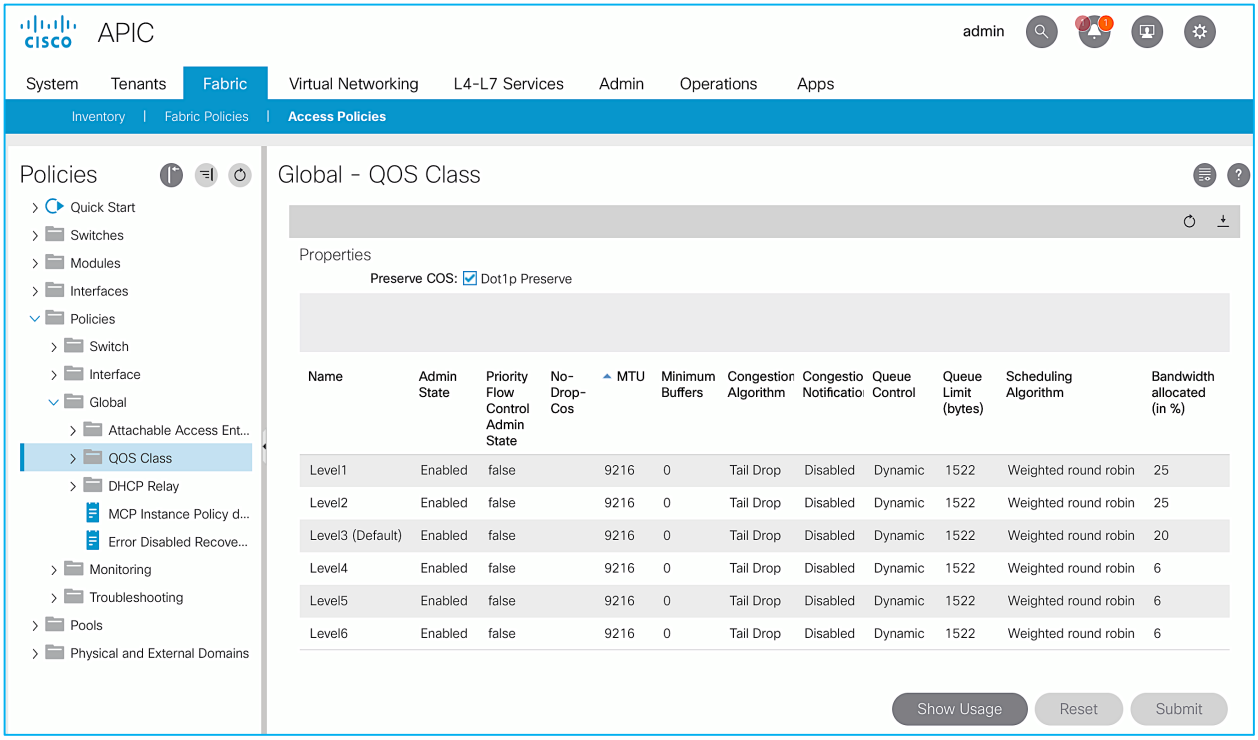

Figure 13.
ACI levels and associated QoS parameters

Each HyperFlex traffic type is allocated a percentage of the bandwidth based on the QoS level it is mapped to. The bandwidth percentage is the **minimum** bandwidth guaranteed to that QoS class **during congestion**. It can get a higher percentage of the bandwidth depending on the traffic pattern and load on the fabric, especially if there is no congestion. ACI uses WRR to ensure fair sharing of the fabric bandwidth. In this design, the number of QoS classes and the bandwidth allocation percent per class are configured to be the same as that of the Cisco HyperFlex UCS domain. Also note that, by default, all QoS levels in ACI support Jumbo Frames.

The bandwidth allocation for the QoS classes in Cisco UCS and ACI fabrics should match when possible. When there is congestion, both fabrics guarantee the minimum bandwidth specified by the bandwidth allocation percent in the QoS policies for that fabric. However, the UCS and ACI fabrics are separate fabrics that operate independently to provide QoS. When both fabrics experience congestion and the bandwidth allocation percentages are the same, the minimum bandwidth guaranteed for a given class of traffic is the same across both fabrics. However, if the allocations percentages are different, it is possible under certain circumstances for the guaranteed minimum across both fabrics to be the lesser of the two. It is also likely that if there is a mismatch, it is on the ACI fabric side because it is a shared fabric for all traffic in a data center. ACI also uses fabric-wide QoS classes, meaning that the parameters for a given QoS apply to all interfaces and links in the fabric. Therefore, it is more likely that you will use higher bandwidth percentages in ACI to accommodate the needs of other endpoints that attach to the same fabric. During congestion, this situation can result in ACI fabric sending more traffic than what the Cisco UCS platform can handle, even though both fabrics meet their minimum bandwidth guarantees. The Cisco UCS platform will drop the excess traffic but it will still meet its minimum bandwidth guarantees. In this scenario, the minimum bandwidth that can be guaranteed across both fabrics will be that of the UCS platform, or the lesser of the two bandwidth allocations.

The key takeaway here is that the minimum bandwidth guaranteed for a QoS class should match between fabrics when possible. If it cannot be matched, then for a given class of traffic, the guaranteed minimum bandwidth will be the lower of the two when both fabrics experience congestion.

It is also important that the guaranteed minimum bandwidth percentage be as accurate as possible so that traffic in that class receives the QoS it needs during congestion. It is particularly important for critical traffic such as HyperFlex storage data traffic. To ensure the accuracy of the bandwidth allocation and other QoS policies, both fabrics should be monitored on an ongoing basis to understand the traffic patterns and bandwidth requirements.

You also can design and size the Cisco ACI and UCS fabrics to prevent oversubscription by using higher-speed links and components capable of handling the network load. You should still monitor traffic on an ongoing basis so that you can upgrade the fabric if the load increases.

CoS preservation

When the ACI fabric receives traffic, it can choose to preserve, modify, or discard the CoS value in the 802.1p frame header. Enabling CoS preservation features tells the fabric to preserve the original CoS so that traffic exiting the fabric has the same CoS value as the traffic entering the fabric. By default, ACI discards the CoS value in the incoming traffic.

In ACI, any traffic received by the fabric is first classified and then encapsulated in an internal Virtual Extensible LAN (VXLAN) header for forwarding through the fabric. The received traffic is classified to an ACI QoS level. The QoS level and the CoS value associated with the traffic is then encoded in the outer DSCP field of the internal VXLAN header. By default, ACI classifies all incoming traffic to a default QoS level (**level 3**). ACI also drops the original CoS (and the IEEE 802.1Q header) when it encapsulates the frame in VXLAN and maps QoS level 3 to a CoS value of "0" and then these values are encoded in the outer DSCP field. CoS 0 is also used to mark the 802.1p field in the outer MAC header of the internal VXLAN header. This means that, by default, ACI will reset the CoS on HyperFlex traffic to CoS 0 as traffic enters the fabric from a Cisco HyperFlex UCS domain. Figure 14 shows the default ACI behavior on ingress leaf switches as it receives traffic from a Cisco HyperFlex UCS domain.

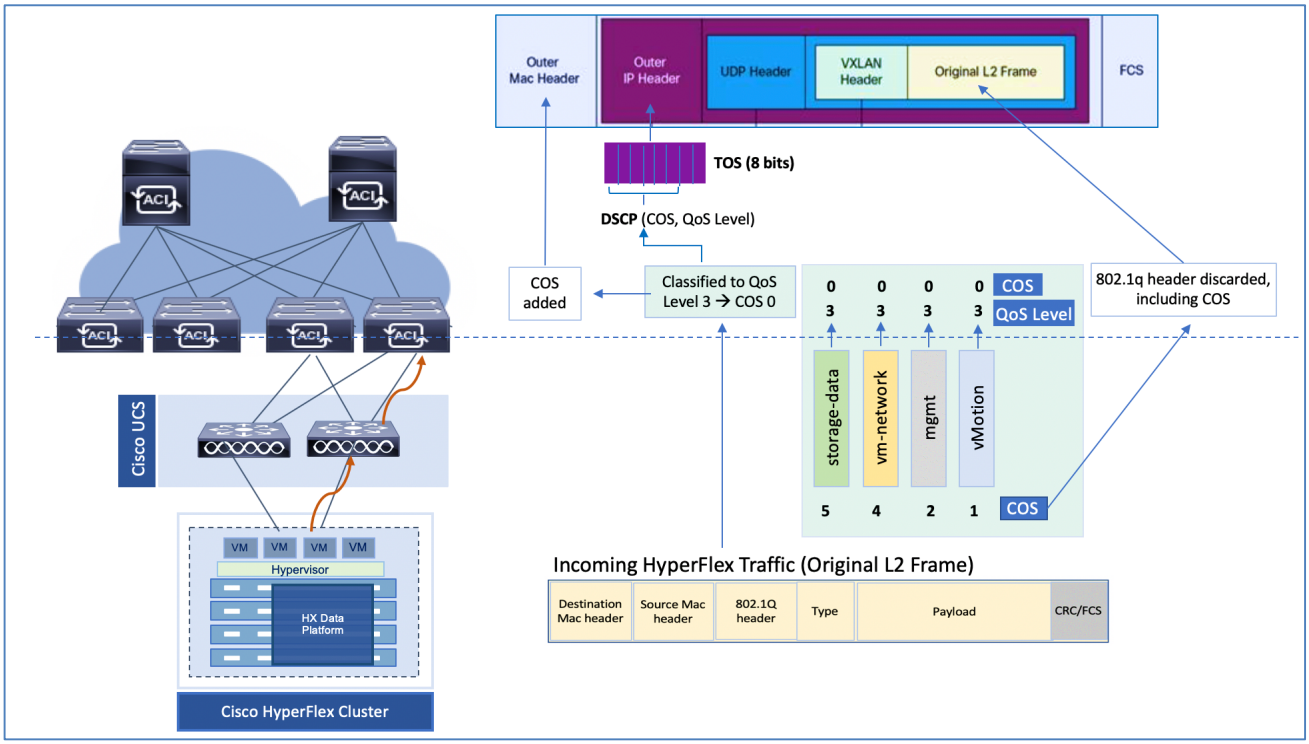

Figure 14.
Default ACI behavior on ingress switches as it receives traffic from Cisco HyperFlex UCS domain

On the egress side, as traffic leaves the fabric, ACI uses the encoded DSCP value in the internal VXLAN header to derive the CoS value. This value is then used to mark the 802.1p field in the outgoing traffic. By default, all outgoing traffic will be marked as CoS 0. In a HyperFlex deployment, this will result in all HyperFlex traffic leaving the fabric to be marked as CoS 0, including storage data traffic. Figure 15 shows the default ACI behavior on egress leaf switches as traffic exits the ACI fabric.

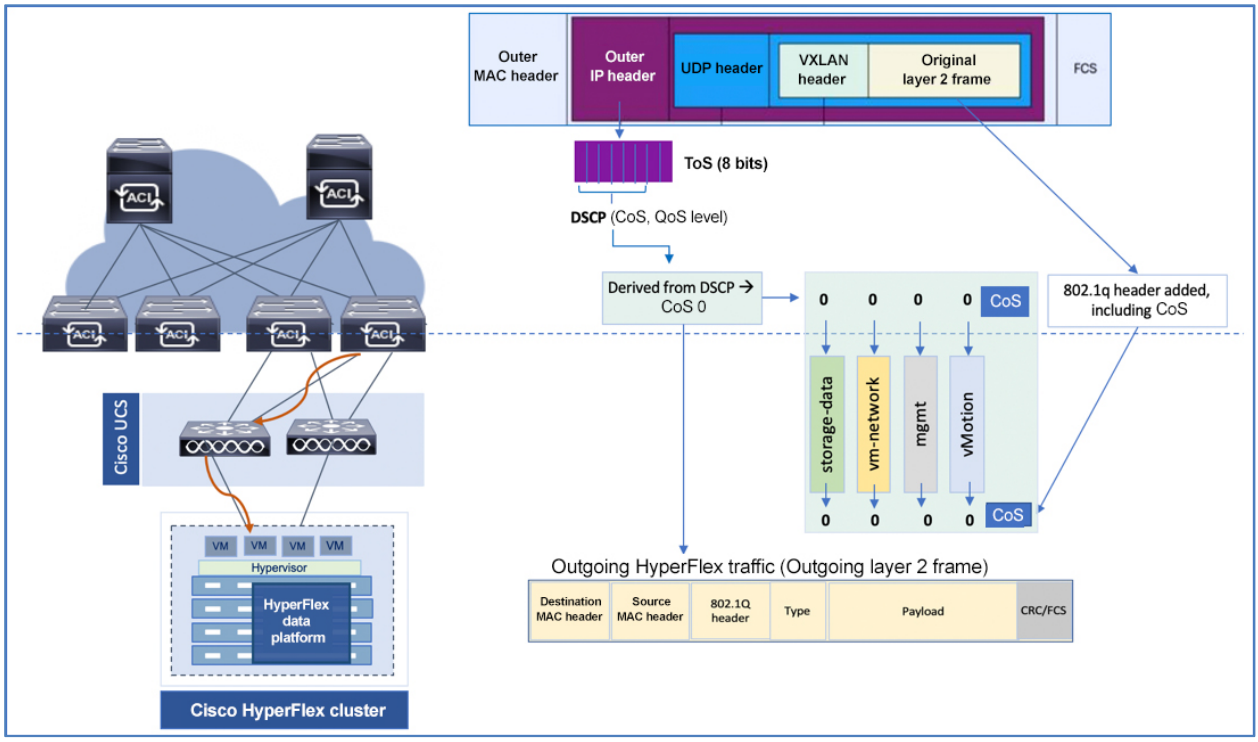

Figure 15.
Default ACI behavior on egress leaf switches as traffic leaves ACI fabric

In the receiving Cisco UCS domain, the HyperFlex CoS 0 traffic is classified into the best-effort class. The HyperFlex installer uses a default or **normal** MTU for the best-effort class. On Cisco UCS 6200 and 6300 Fabric Interconnect models, **normal** MTU is 1500B. This MTU can result in HyperFlex storage and vMotion traffic that uses Jumbo Frames to get dropped. During congestion, treating HyperFlex storage traffic as a best-effort service can also destabilize the storage cluster and lead to poor storage performance.

To prevent this occurrence, when HyperFlex clusters are connected to an ACI fabric, the CoS in the incoming traffic should be preserved across the fabric using one of the following features:

- **Dot1p Preserve** feature (for ACI single-site or Multi-Pod deployments)
- **COS to DSCP Translation** policy (for ACI Multi-Pod deployments)

Note that if both the **Dot1p Preserve** feature and the **COS to DSCP Translation** policy are configured, the **COS to DSCP Translation** policy takes precedence because they both modify the same DSCP field in the internal VXLAN header.

You can use the **Dot1p Preserve** feature when HyperFlex (standard or stretched) clusters are connected to a single-site ACI fabric. However, for a stretched cluster extended across an ACI Multi-Pod fabric, you can use either the **Dot1p Preserve** feature or a **COS to DSCP Translation** policy to preserve CoS. A **COS to DSCP Translation** policy is required in some environments; the following section gives more details.

Dot1p Preserve

When the **Dot1p Preserve** feature is enabled, ACI preserves the CoS value in the incoming traffic by encoding the CoS (and QoS level) in the outer DSCP field of the internal VXLAN header that ACI uses for forwarding. The encoding is done by the leaf switches that first receive the traffic. ACI derives the encoded DSCP value based on the received CoS and QoS level that the traffic maps to, and it cannot be modified. The egress leaf switch uses this DSCP value to derive the CoS value and re-mark the traffic as it exits the fabric.

Dot1p Preserve is a fabric-wide policy that is enabled as shown in Figure 16.

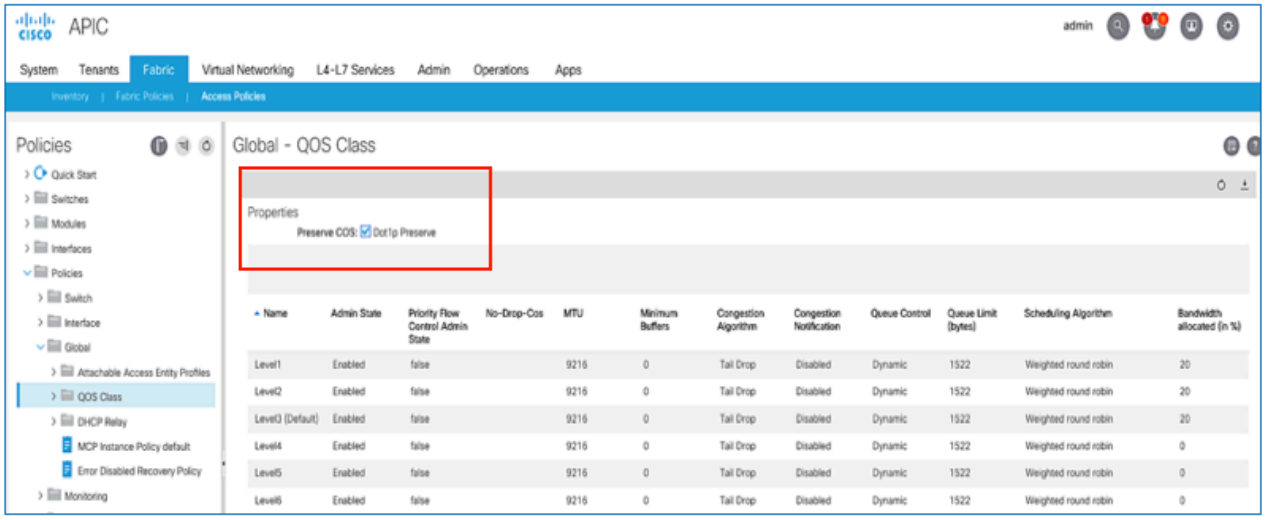

Figure 16.
Enabling Dot1p Preserve

HyperFlex deployments can use this feature to preserve CoS as long as there is no risk of the encoded DSCP getting modified along the path. If the DSCP is modified, the traffic leaving the fabric is marked using an incorrect CoS that was derived from the modified DSCP. If QoS is enabled in the ACI fabric, the QoS level will also be derived from the modified DSCP, and the quality of service the traffic receives within the fabric can be affected as well. Note that spine switches use DSCP to derive QoS level for traffic received from external networks but uses the outer CoS field for traffic received from leaf switches. Egress leaf switches also use DSCP to determine both the QoS level and the CoS value for traffic exiting the fabric.

The ACI fabric does not typically modify the encoded DSCP. Therefore, any standard or HyperFlex stretched cluster connected to a single-site ACI fabric can use this feature to preserve CoS. It can also be used for an ACI Multi-Pod fabric if the DSCP is not modified in the IPN. However, if there is a risk of the DSCP getting modified in the IPN, you should use the **COS to DSCP Translation** policy discussed in the next section.

Table 5 shows the encoded DSCP value for the different types of HyperFlex traffic the ACI fabric receives.

Table 5. Encoded DSCP value for HyperFlex traffic

HyperFlex traffic type	Source EPG	QoS level	Encoded DSCP (ACI-derived)
HyperFlex storage data	HXV-CL1-StorData	Level 1	42
Virtual-machine network traffic	HXV-A-App_EPG, HXV-A-Web_EPG, ...	Level 2	33
HyperFlex management	HXV-IB-MGMT_EPG	Level 4	19
vMotion	HXV-vMotion_EPG	Level 5	12

The ACI derived DSCP values are based on the CoS value of the incoming traffic and the QoS level it maps to. For a comprehensive list of ACI derived DSCP values in ACI 4.0 and later, refer to Appendix B.

CoS-to-DSCP translation policy

When you use a **COS to DSCP Translation Policy**, ACI preserves the CoS value in the incoming traffic by encoding the CoS (and QoS level) in the outer DSCP field of the internal VXLAN header that ACI uses for forwarding. This feature is the same as the **Dot1p Preserve** feature. However, unlike the **Dot1p Preserve** feature, the encoded DSCP value is specified by the policy and not derived by ACI. Also, the DSCP marking based on the **COS to DSCP Translation Policy** is done by the spine switches before the traffic is forwarded to the IPN and not by the ingress leaf switches.

A **COS to DSCP Translation** policy is necessary when administrators need the flexibility to specify the DSCP value that a given CoS and QoS level maps to. For example, if the ACI-derived DSCP value is in use by other traffic on the IPN, administrators can change it using a **COS to DSCP Translation** policy.

A **COS to DSCP translation** policy may also be necessary if there is a risk of IPN routers or switches modifying the CoS value in the traffic sent across an IPN. ACI typically uses the outer CoS field in the internal VXLAN header to determine the QoS level for a packet. However, in an ACI Multi-Pod environment, If the CoS value gets modified in the IPN, the spine switches will map the received traffic to an incorrect QoS level. If a **COS to DSCP Translation** policy is used, the spine switches will use the policy and the DSCP in the received traffic to determine the QoS level. Leaf switches will also use the policy for traffic it receives from the fabric to determine the QoS level and to derive the original CoS so that it can be used to re-mark the traffic correctly as it leaves the fabric.

Table 6 shows the user-specified DSCP values that were used in this design to encode the CoS and QoS level for the different HyperFlex traffic types; these values are specified in the CoS-to-DSCP translation policy. The different types of HyperFlex traffic the ACI fabric receives, original CoS values, and the EPGs and QoS levels they map to are also shown. The ACI-derived DSCP value based on the incoming CoS and the QoS level it maps to are also provided for comparison purposes.

Table 6. User-specified DSCP values used for HyperFlex traffic

HyperFlex traffic type	Original CoS (from Cisco UCS domain)	EPG	QoS class	ACI-derived DSCP	User-specified DSCP in the policy
HyperFlex storage data	COS 5 (Platinum)	HXV-CL1-StorData	Level 1	42	CS4 (32)
Virtual-machine network traffic	COS 4 (Gold)	HXV-A-App_EPG, HXV-A-Web_EPG, ...	Level 2	33	Application-dependent AF31 (26)
HyperFlex management	COS 2 (Silver)	HXV-IB-MGMT_EPG	Level 4	19	CS3 (24)
vMotion	COS 1 (Bronze)	HXV-vMotion_EPG	Level 5	12	AF11 (10)

Note: DSCP values 57–63 are reserved for internal use and cannot be used. The DSCP values are user specified so customers can change them as needed to suit the needs of their deployment.

The **COS to DSCP translation** policy is specified in the **Infra** Tenant because it affects the internal VXLAN headers that ACI uses for forwarding across the fabric. The DSCP values specified for the different HyperFlex traffic types are defined as shown in Figure 17.

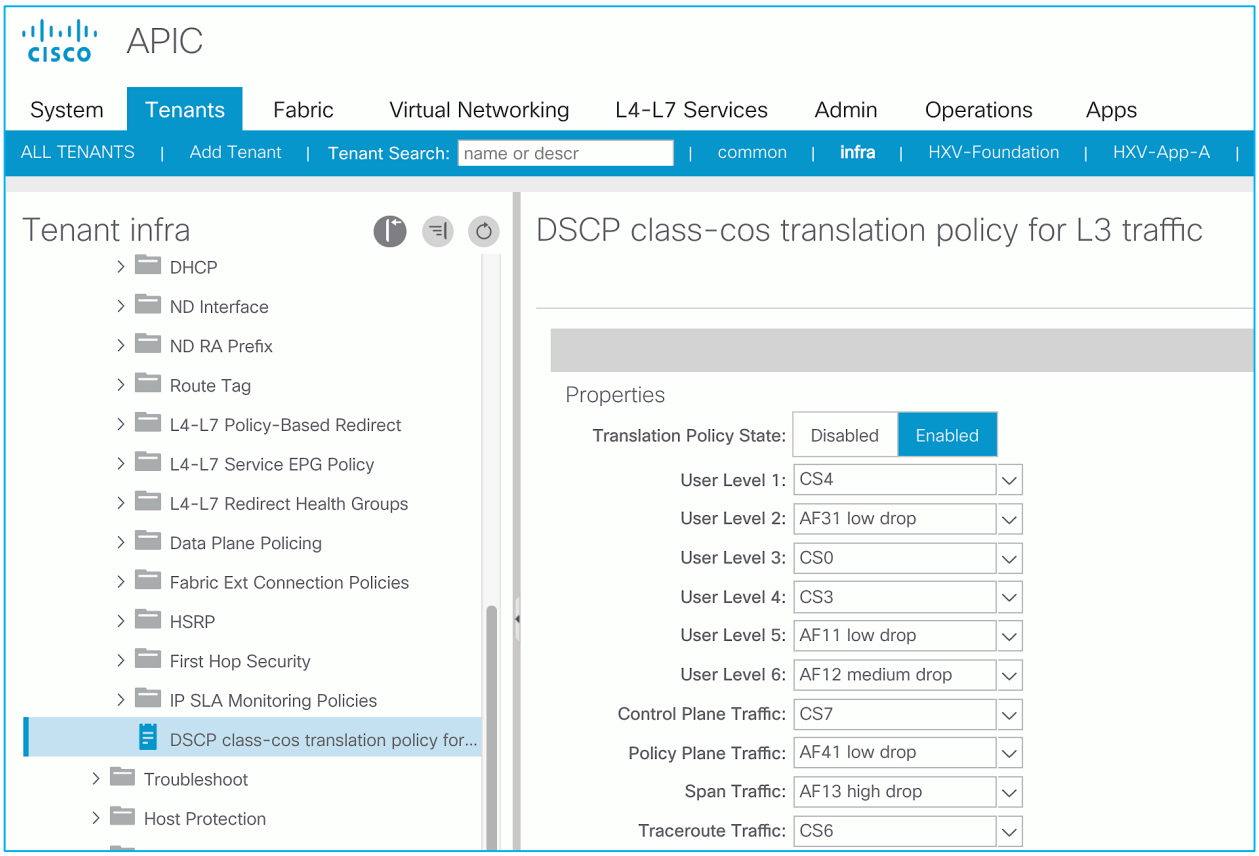

Figure 17.
DSCP values for HyperFlex traffic types

Note that if you use a **COS to DSCP Translation** policy, CoS translation using a **Custom QoS** policy is not supported.

Inter-Pod Network QoS

Inter-Pod network (IPN) in an ACI Multi-Pod fabric provides connectivity between HyperFlex nodes in a HyperFlex stretched cluster. To implement QoS in the IPN for HyperFlex traffic, a sample IPN QoS configuration is provided below. The QoS configuration is based on the Cisco Nexus 93180YC-EX model of IPN switches. The primary goal is to ensure that HyperFlex storage and data traffic receives the guaranteed minimum bandwidth during periods of congestion. You can use the same configuration on the different IPN switches that connect to spine switches in each pod. The bandwidth percentages reflect the values used in the ACI fabric and Cisco UCS domains; you should adjust the QoS configuration and policies as needed to meet the needs of your organizations.

IPN QoS configuration

Create class maps to match the markings configured on the APIC.

```
class-map type qos match-all ACI-QoS-Level1
  match dscp 32
class-map type qos match-all ACI-QoS-Level2
  match dscp 26
class-map type qos match-all ACI-QoS-Level4
  match dscp 24
class-map type qos match-all ACI-QoS-Level5
  match dscp 10
class-map type qos match-all ACI-QoS-Level3-6
  match dscp 0, 12
class-map type qos match-all ControlAndPolicyTraffic
  match dscp 56,34
class-map type qos match-all SpanTraffic
  match dscp 14
class-map type qos match-all TracerouteTraffic
  match dscp 48
```

Create a policy to map the traffic to a QoS group.

```
 class ControlAndPolicyTraffic
   set qos-group 7
 class ACI-QoS-Level1
   set qos-group 6
 class ACI-QoS-Level2
   set qos-group 4
 class ACI-QoS-Level4
   set qos-group 3
 class ACI-QoS-Level5
   set qos-group 2

class ACI-QoS-Level3-6
```

Create a policy to map the traffic to a QoS group.

```
    set qos-group 0
  class SpanTraffic
    set qos-group 1
  class TracerouteTraffic
    set qos-group 5
```

Configure Queuing for the QoS group.

```
policy-map type queuing HXV-ACI-8q-out-policy-To-IPN
  class type queuing c-out-8q-q7
    bandwidth percent 10
  class type queuing c-out-8q-q6
    bandwidth percent 25
  class type queuing c-out-8q-q5
    bandwidth remaining percent 1
  class type queuing c-out-8q-q4
    bandwidth percent 25
  class type queuing c-out-8q-q3
    bandwidth percent 6
  class type queuing c-out-8q-q2
    bandwidth percent 6
  class type queuing c-out-8q-q1
    bandwidth remaining percent 1
  class type queuing c-out-8q-q-default
    bandwidth percent 20

system qos
  service-policy type queuing output HXV-ACI-8q-out-policy-To-IPN
```

Associate the interfaces connected to the spine switch with the service policy.

```
interface Ethernet1/49.4
  description To Spine Switch in ACI Pod
  service-policy type qos input HXV-ACI-Traffic-Classification

interface Ethernet1/50.4
  description To Spine Switch in ACI Pod
  service-policy type qos input HXV-ACI-Traffic-Classification
```

Appendix A—Cisco UCS CLI-Based QoS configuration

Cisco UCS QoS configuration

class-map type qos match-all class-gold
  match cos 4
class-map type qos match-all class-bronze
  match cos 1
class-map type qos match-all class-silver
  match cos 2
class-map type qos match-all class-platinum
  match cos 5

class-map type queuing class-gold
  match qos-group 3
class-map type queuing class-bronze
  match qos-group 5
class-map type queuing class-silver
  match qos-group 4
class-map type queuing class-platinum
  match qos-group 2

policy-map type qos system_qos_policy
  class class-fcoe
  set qos-group 1
  class class-platinum
  set qos-group 2
  class class-silver
  set qos-group 4
  class class-bronze
  set qos-group 5
  class class-gold
  set qos-group 3

policy-map type queuing system_q_in_policy
  class type queuing class-fcoe
  bandwidth percent 32
  class type queuing class-platinum
  bandwidth percent 25
  class type queuing class-gold
  bandwidth percent 25
  class type queuing class-silver
  bandwidth percent 6
  class type queuing class-bronze
  bandwidth percent 6
  class type queuing class-default
  bandwidth percent 6

Cisco UCS QoS configuration

policy-map type queuing system_q_out_policy
  class type queuing class-fcoe
  bandwidth percent 32
  class type queuing class-platinum
  bandwidth percent 25
  class type queuing class-gold
  bandwidth percent 25
  class type queuing class-silver
  bandwidth percent 6
  class type queuing class-bronze
  bandwidth percent 6
  class type queuing class-default
  bandwidth percent 6

policy-map type queuing ingress_queuing_852908
  class type queuing class-default
  bandwidth percent 100
  shape 40000000 kbps 10240
policy-map type queuing org-root/org-HXV-Org1/ep-qos-gold
  class type queuing class-default
  bandwidth percent 100
  shape 40000000 kbps 10240
policy-map type queuing org-root/org-HXV-Org1/ep-qos-bronze
  class type queuing class-default
  bandwidth percent 100
  shape 40000000 kbps 10240
policy-map type queuing org-root/org-HXV-Org1/ep-qos-silver
  class type queuing class-default
  bandwidth percent 100
  shape 40000000 kbps 10240
policy-map type queuing org-root/org-HXV-Org1/ep-qos-platinum
  class type queuing class-default
  bandwidth percent 100
  shape 40000000 kbps 10240

class-map type network-qos class-gold
  match qos-group 3
class-map type network-qos class-bronze
  match qos-group 5
class-map type network-qos class-silver
  match qos-group 4
class-map type network-qos class-platinum
  match qos-group 2

Cisco UCS QoS configuration

policy-map type network-qos system_nq_policy
  class type network-qos class-fcoe
  pause no-drop
  mtu 2158
  class type network-qos class-platinum
  mtu 9216
  pause no-drop
  class type network-qos class-silver
  class type network-qos class-bronze
  mtu 9216
  class type network-qos class-gold
  mtu 9216
  class type network-qos class-default

system qos
  service-policy type qos input system_qos_policy
  service-policy type queuing input system_q_in_policy
  service-policy type queuing output system_q_out_policy
  service-policy type network-qos system_nq_policy

Appendix B—ACI-derived DSCP

Table 7 shows the ACI-derived DSCP for CoS preservation for a given CoS value and QoS level of the packet. This data is for APIC Release 4.0 and later. Some DSCP values (3–7, 57–63) are reserved. Note that this table is subject to change in the future.

Table 7. ACI-derived DSCP for CoS preservation for a given CoS value and QoS level of packet

Incoming traffic level based on EPG QoS policy	Incoming Dot1p value in packet	ACI-derived DSCP value
LEVEL3	0	0
LEVEL2	0	1
LEVEL1	0	2
RESERVED	X	3
RESERVED	X	4
RESERVED	X	5
RESERVED	X	6
RESERVED	X	7
LEVEL3	1	8
LEVEL2	1	9
LEVEL1	1	10
LEVEL4	1	11
LEVEL5	1	12
LEVEL6	1	13
LEVEL2	7	14
LEVEL1	7	15
LEVEL3	2	16
LEVEL2	2	17
LEVEL1	2	18
LEVEL4	2	19
LEVEL5	2	20
LEVEL6	2	21
LEVEL4	0	22
LEVEL6	7	23
LEVEL3	3	24
LEVEL2	3	25
LEVEL1	3	26

Incoming traffic level based on EPG QoS policy	Incoming Dot1p value in packet	ACI-derived DSCP value
LEVEL4	3	27
LEVEL5	3	28
LEVEL6	3	29
LEVEL4	7	30
LEVEL3	4	32
LEVEL2	4	33
LEVEL1	4	34
LEVEL4	4	35
LEVEL5	4	36
LEVEL6	4	37
LEVEL5	0	38
LEVEL3	5	40
LEVEL2	5	41
LEVEL1	5	42
LEVEL4	5	43
LEVEL5	5	44
LEVEL6	5	45
LEVEL5/IFC RESERVED	7	46
LEVEL3	6	48
LEVEL2	6	49
LEVEL1	6	50
LEVEL4	6	51
LEVEL5	6	52
LEVEL6	6	53
LEVEL6	0	54
LEVEL3	7	56
RESERVED		57, 58
RESERVED	1	59
RESERVED	0	60
RESERVED	7	61
RESERVED	7	62
RESERVED	7	63

References

1. Cisco APIC QoS: https://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/kb/Cisco-APIC-and-QoS.html - id_40416
2. Tuning Guidelines for Cisco UCS Virtual Interface Cards: https://www.cisco.com/c/dam/en/us/products/collateral/interfaces-modules/unified-computing-system-adapters/vic-tuning-wp.pdf
3. Cisco ACI Design Guide White Paper: https://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/white-paper-c11-737909.html - _Toc6452890
4. Intelligent Buffer Management on Cisco Nexus 9000 Series Switches White Paper: https://www.cisco.com/c/en/us/products/collateral/switches/nexus-9000-series-switches/white-paper-c11-738488.html
5. Operating Cisco HyperFlex HX Data Platform Stretch Clusters: https://www.cisco.com/c/dam/en/us/products/collateral/hyperconverged-infrastructure/hyperflex-hx-series/operating-hyperflex.pdf
6. Cisco HyperFlex 3.5 Stretched Cluster with Cisco ACI 4.0 Multi-Pod Fabric Design Guide: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/hx_35_vsi_aci_multipod_design.html
7. Cisco HyperFlex 3.5 Stretched Cluster with Cisco ACI 4.0 Multi-Pod Fabric Deployment Guide: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/hx_35_vsi_aci_multipod.html

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **https://www.cisco.com/go/offices**.

Printed in USA 219272-1 09/19