Deployment of Active - Active High Availability system using Ethernet RDMA

Configuring Layer 2 shared Inter-Switch Link (ISL) in IBM® HyperSwap®

Table of contents

Overview

Challenges

1. IBM® HyperSwap® was only supported in Fibre Channel environment, which is a costly solution.
2. Multiple ISLs between two sites are costly as well as difficult to manage.
3. Using a single fault tolerant ISL mandates sharing between multiple data traffics. Shared ISL is congestion prone.

Solutions

1. The IBM HyperSwap solution is now qualified for pure Ethernet data centers in release 8.5.2.0, making it a more cost-effective option.
2. Implementing a shared ISL IBM HyperSwap solution ensures that a single link is shared between multiple data traffics, which is a more cost-effective and manageable solution.
3. The document provides an end-to-end solution for the shared ISL that includes measures to prevent congestion.

This paper aims to assist in the deployment of IBM FlashSystem™ as an IBM HyperSwap function in an Ethernet environment. It offers insights into the use of Shared Inter-Switch Link (ISL) in IBM HyperSwap configuration and provides a comprehensive guide on the complete configuration process, including details on the solution topology.

Why Ethernet HyperSwap

The IBM HyperSwap is a clustered, high-availability solution that operates in an active-active mode. It is available on systems that support multiple I/O groups.

Previously, IBM FlashSystem-only supported Fibre Channel based IBM HyperSwap configurations. With the introduction of FlashSystem node-to-node communication support over Ethernet/IP connectivity, starting with the 8.2.1 release, it is now possible to utilize Ethernet connectivity for dual-site active-active solutions, enabling efficient and reliable data access and management across multiple sites.

With the release of version 8.3.1, IBM FlashSystem has introduced Ethernet-based IBM HyperSwap solutions based on SCORE/Request for Price Quotation (RPQ) requests.

With the release of version 8.5.2.0, Ethernet-based IBM HyperSwap configurations are now qualified for general availability for iWARP connectivity.

This new solution allows Ethernet data centers to deploy IBM HyperSwap without relying on Fibre Channel interconnect. Therefore, the absence of Fibre Channel over IP (FCIP) routers reduces the Total Cost of Ownership (TCO) for IBM HyperSwap solutions deployed in an Ethernet environment.

Ethernet-based IBM HyperSwap is suitable for synchronous replication between near distance disaster recovery sites in 3-site coordinated replication solutions. This capability allows for the deployment of 3-site coordinated replication in an Ethernet-only environment without the need for Fiber Channel infrastructure.

Inter-Switch Link

ISL is a connectivity link between two switching fabrics. The term "Shared ISL" refers to the sharing of ISL bandwidth between different types of network traffic. In a shared ISL network configuration, various network traffic types can flow through the same ISL simultaneously. This means that different types of network traffic can share the same ISL bandwidth, allowing for maximum efficiency and utilization of resources.

This document outlines the configuration steps for Ethernet-based IBM HyperSwap solutions, which can be used for FlashSystem nodes. While the example provided in this paper utilizes an IBM V7000 (2076-624) enclosure for the configuration, the steps and guidelines outlined here are applicable to other IBM FlashSystems as well.

HyperSwap® solution with shared ISL network configuration can be used where round-trip time (RTT) is up to 1 millisecond.

Prerequisites

The following prerequisites are to be considered before deployment of Ethernet HyperSwap.

Initial Setup considerations

Prior to the deployment of Ethernet HyperSwap, it is essential to plan the number of Remote Direct Memory Access (RDMA) capable Ethernet adapters in the FlashSystems to support node-to-node and host-to-node connectivity.

Required hardware & software components

Refer to Planning network cable connections on ibm.com for supported iWARP adapter, cables, and Small Form-factor Pluggable (SFP) used for IBM HyperSwap.

It is important to note that Ethernet-based IBM HyperSwap is only certified for Layer 2 networks and iWARP technology. In scenarios where a Layer 3 network is required, certain technologies such as Virtual Extensible LANs (VXLANs) may be utilized to enable overlaying Layer 2 (L2) networks on a Layer 3 underlay.

Network Requirements

To create a shared ISL configuration for Ethernet HyperSwap, a minimum of two switches are required on each site that will be ISL'ed to switches on the other site. Additionally, a Data Center Bridging Capability Exchange (DCBx) capable switch is required to fulfil the requirement.

The following table lists the IEEE standard required in the switch:

IEEE Standard	Description
802.1Q	VLAN
P802.1p	CoS - Traffic Class Expediting and Dynamic Multicast Filtering
802.1DC	Quality of Service Provision by Network Systems

802.1Qaz	Enhanced Transmission Selection for Bandwidth Sharing Between Traffic Classes
802.1Qbb	Priority-based Flow Control
802.1Qaz 2011	Data Center Bridging Exchange

Table 1: Switch requirements as per IEEE standard.

The preceding standards help configure switches to handle congestion scenarios caused by different types of traffic flowing through the same ISL.

The following figures 1 and 2 are examples of traffic with and without congestion handling. In figure 1, different colors represent multiple traffics. Blue is critical, i.e., Node-to-node traffic and green being host-to-node traffic.

When host-to-node traffic increases and consumes high amounts of bandwidth, it overwhelms other traffic. This inflicts a major bottleneck in the network due to congestion and affects other traffic streaming through the same ISL. This situation might cause outage. The same is illustrated in figure 1.

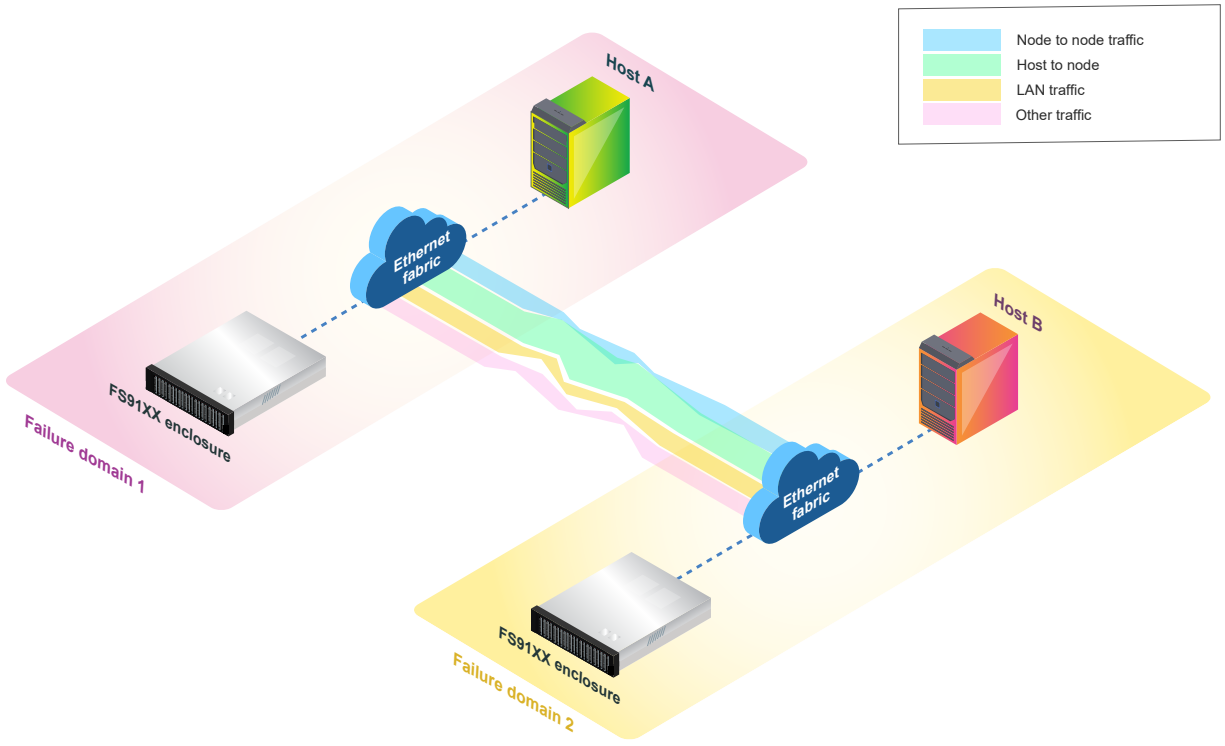

Figure 1: Data traffic with network congestion and traffic impacting each other.

To avoid congestion and bottlenecks in the IBM HyperSwap shared ISL configuration, traffic marking, classification, and network shaping techniques can be employed. By utilizing these techniques, all traffic can flow within the same ISL without causing bottlenecks or impacting other traffic, as demonstrated in Figure 2.

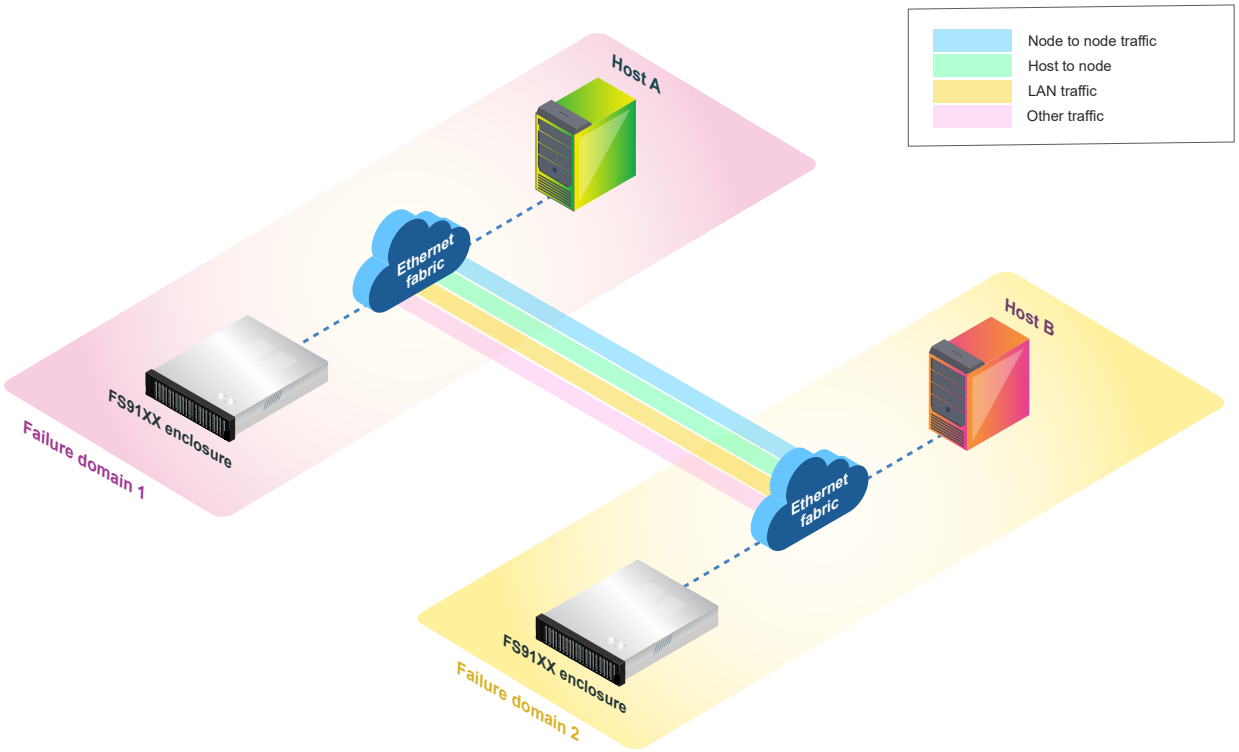

Figure 2: Data traffic with network congestion handling.

HyperSwap solution reference architecture

This whitepaper describes the reference architecture for an IBM HyperSwap solution. The lab environment was configured to simulate a production-ready environment.

The configuration consists of two failure domains or sites, separated by not more than 300 meters, with IBM FlashSystem enclosures installed on each site. Four Cisco Nexus 3232C 100 Gb switches are used to connect the IBM FlashSystem nodes and hosts, using 4x25 Gb splitter cables. Each site has two switches installed, and all FlashSystem nodes and hosts are connected to each switch to achieve full redundancy.

The connection between the two sites is accomplished through 2x25 Gb ISL links, shown as blue and green lines in Figure 3, for node-to-node and host-to-node communication respectively. IP Quorum is configured between the two IBM FlashSystem enclosures through a separate management network, which is considered as a third site or failure domain.

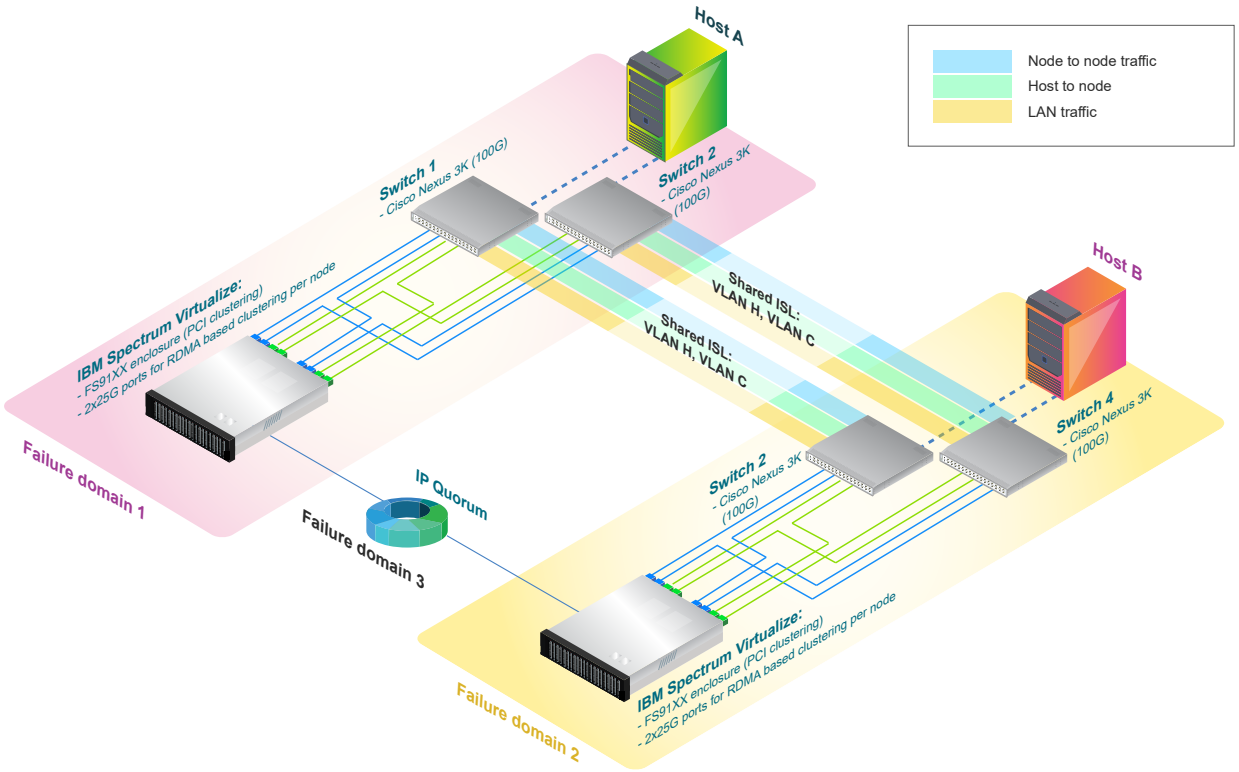

Figure 3: Logical view of network connectivity.

Note: The Blue lines indicate node-to-node communication links, and the green lines indicate host-to-node connectivity links. The dotted blue line represents the IP quorum, accessible from both sites.

Legends	Description
VLAN H	Used for host-to-node connectivity/traffic
VLAN C	Used for node-to-node connectivity/traffic

Table 2: Legends used in Figure 3.

Detailed physical connectivity

The physical connectivity between the 100G switches and FlashSystem Ethernet ports in the two failure domains is illustrated in Figure 4, providing a detailed diagram of port-to-port connectivity links with different colors. The diagram also outlines the recommended configuration of the management network, using onboard 10G ports, which are highlighted in green.

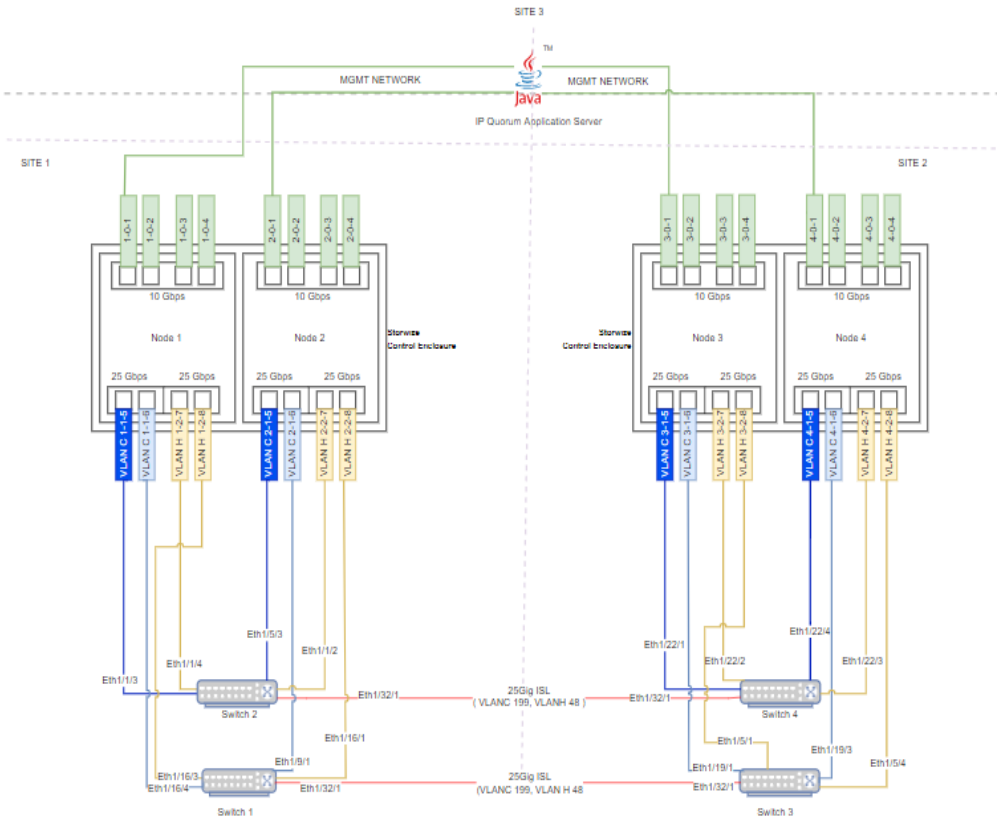

Figure 4: Physical network connectivity diagram for Shared ISL HyperSwap® configuration.

Legends	Description
--* or can be read as N-S-P	First * indicates Node number Second * indicates PCI Slot number Third * indicates Port number

-0- (example 1-0-1, 1-0-2, 3-0-1, 4-0-4)	Each port is an onboard 10G port (PCI slot 0)
VLAN C/H *-*-* (e.g VLAN C 2-1-5, VLAN H 3-2-7)	C indicates port is used for node-to-node connectivity H indicates port is used for host-to-node connectivity First * indicates **N**ode number Second * indicates PCI **S**lot number Third * indicates **P**ort number

Table 3: Legends used in Figure 4.

Link	**Description**
VLAN C 1-1-6, VLAN C 2-1-6,	Node-to-node link from switch 1
VLAN C 1-1-5, VLAN C 2-1-5,	Node-to-node link from switch 2
VLAN C 3-1-6, VLAN C 4-1-6	Node-to-node link from switch 3
VLAN C 3-1-5, VLAN C 4-1-5	Node-to-node link from switch 4
VLAN C 1-1-5 to Eth1/1/3	Node1 port 5 connected to switch 2
VLAN C 2-1-5 to Eth1/5/3	Node2 port 5 connected to switch 2
VLAN C 1-1-6 to Eth1/16/4	Node1 port 6 connected to switch 1
VLAN C 2-1-6 to Eth1/9/1	Node2 port 6 connected to switch 1

VLAN C 3-1-5 to Eth1/22/1	Node3 port 5 connected to switch 4
VLAN C 4-1-5 to Eth1/22/4	Node4 port 5 connected to switch 4
VLAN C 3-1-6 to Eth1/19/1	Node3 port 6 connected to switch 2
VLAN C 4-1-6 to Eth1/19/3	Node4 port 6 connected to switch 2

Table 4: Connectivity links in Figure 4.

The onboard 10G Ethernet ports are utilized for management connectivity, and they can also serve as Internet Small Computer System Interface (iSCSI) host-to-node connections. To establish IP quorum, a separate network is configured, which is accessible from both sites.

Configuring host-to-node and node-to-node connections via different subnets is a mandatory requirement. Furthermore, it is must to have separate VLANs for host-to-node and node-to-node connections to isolate link bandwidths.

Figure 5 is a rear view of FlashSystem 91xx control enclosure which contains two identical node canisters. One of the control enclosures is in Failure Domain1 and another control enclosure is in Failure Domain2. The top node canister(node1/node3) is inverted above the bottom one(node2/node4); each node canister is bound on each side by a power supply unit.

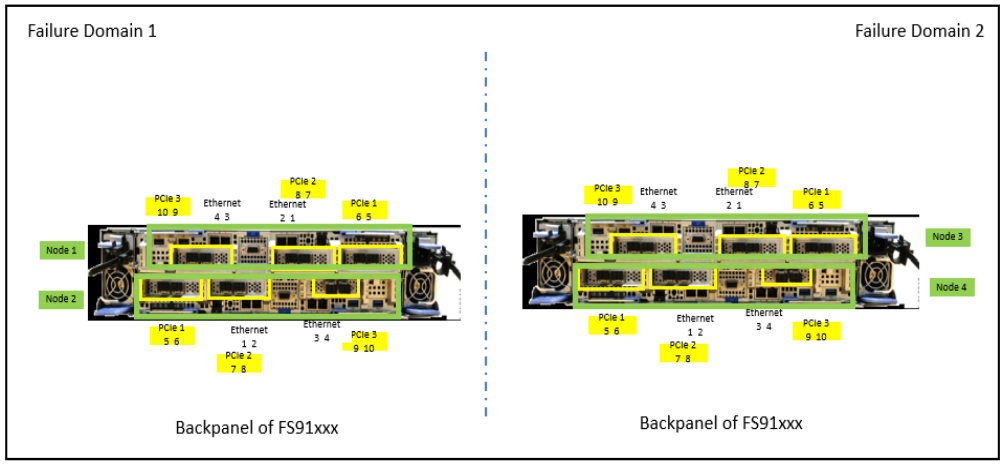

Figure 5: Back Panel of FS91XX in respect of failure domains.

General recommendations for Ethernet-based clustering over RDMA

- Establish inter-node connectivity by connecting identical ports of nodes, such as Node 1's port 4 to Node 2's port 4. These identical ports must run on the iWARP protocol.
- Use different protocols for host-to-node versus node-to-node communication. For example, use iSCSI and Non-Volatile Memory express (NVMe) RDMA over Converged Ethernet (RoCE) for host-to-node connectivity and iWARP for node-to-node connectivity.
- Configure different subnets and VLANs for host-to-node and inter-node connectivity to ensure better network performance and security.
- Use at least two dedicated RDMA-capable Ethernet ports per node for inter-node communication. These ports should be used exclusively for node-to-node traffic and not shared for host-to-node, storage virtualization, or IP replication traffic.

Configuration of Layer 3 switch

Implemented environment

This white paper focuses on the configuration of the Cisco Nexus 3232C Layer 3 Switch, which was utilized for the qualification of IBM HyperSwap solution. Although other switch vendors are available, the specific steps outlined below are tailored for the Cisco Nexus 3232C and may vary for other providers.

In the lab environment, Cisco Nexus 3232C switches with 100 Gb ports were used. These 100 Gb ports can be connected through 4x25 Gb splitter cables, with 25 Gb ends linked to IBM FlashSystem node and host ports through 25 Gb supported SFPs if the splitter cable doesn't have an inbuilt SFP.

Implementation

The HyperSwap® solution requires appropriate Class of service (CoS) marking for traffic such as node-to-node and host-to-host, which can be achieved using the ‘chsystemethernet’ command line interface (CLI) command. When network packets marked with a "class of service" reach the switch, they are assigned specific Quality of Service settings to manage traffic flow and avoid transmission delays. This ensures that the fabric meets the expected service quality for applications and delivers the expected IO performance.

By allocating specific bandwidth to traffic, Quality of Service (QoS) ensures that traffic receives the bandwidth as defined within the solution and prevents congestion scenarios. For instance, if the allocated bandwidth for a traffic is 50%, the switch will limit other traffic from using more bandwidth and overeating the allocated bandwidth. Understanding the requirements of the HyperSwap solution and applying the correct traffic shaping settings ensures that the overall solution avoids network bottlenecks arising due to shared network ISL between different traffics.

The lab environment implementation of the IBM HyperSwap solution involved FS9100 nodes connected to Cisco 3232C switches via a 10 Gb/s speed ISL link between switches, as shown in Figure 6. The IBM FlashSystem management GUI monitoring tab provides a quick overview of the system performance without congestion handling. However, the "Interfaces" section of the graph, specifically the iSCSI extensions for RDMA (iSER) representing the clustering traffic, shows a significant disruption due to the node-to-node traffic flowing through the ISL and getting affected by other traffics using the same ISL, causing a congestion scenario.

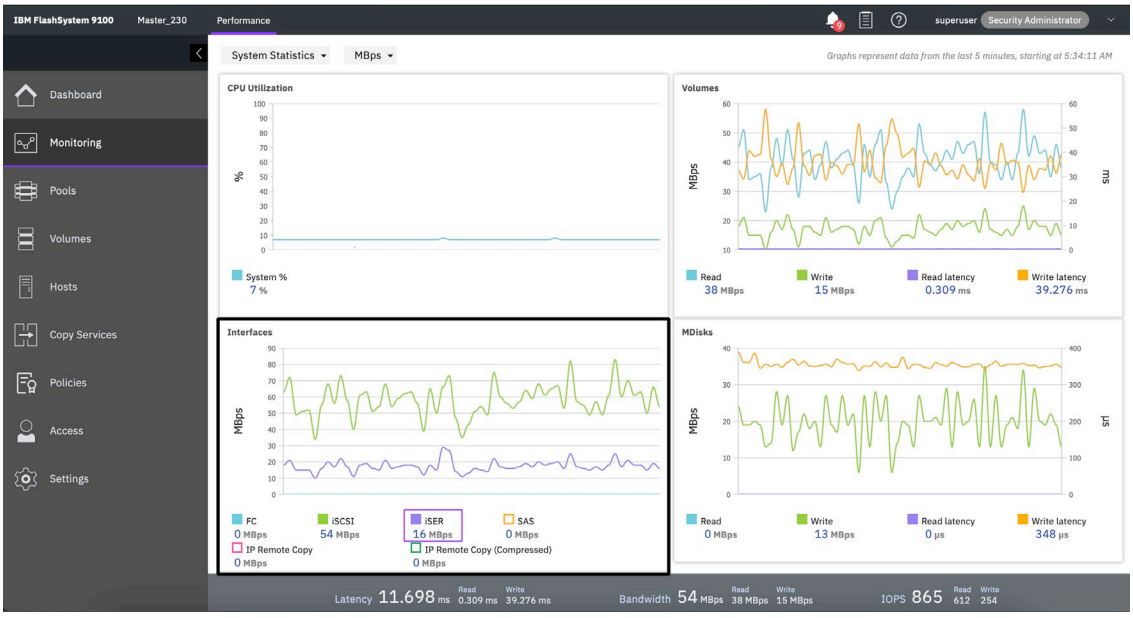

Figure 6: GUI performance overview without congestion handling.

The IBM FlashSystem management GUI monitoring tab in Figure 7 displays the system performance with configured priority flow control. Host-to-node and node-to-node traffics are allocated 70% and 30% bandwidth, respectively. In the "Interfaces" section of the graph, the continuous line for iSER, which represents clustering traffic, indicates smooth flow of node-to-node traffic over the ISL without disruptions caused by other traffics sharing the same ISL.

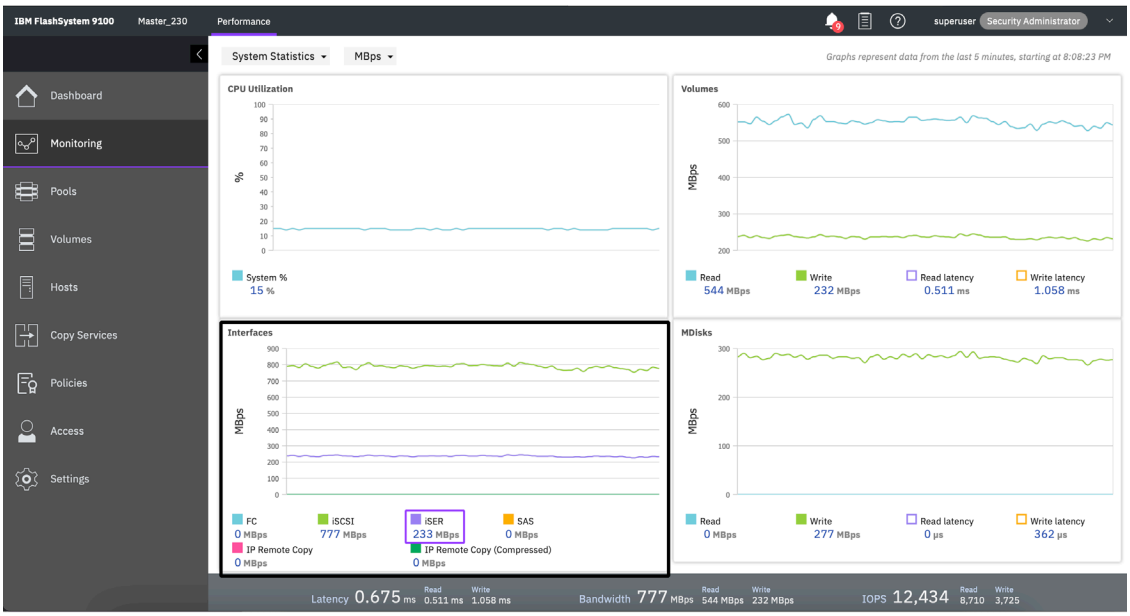

Figure 7: GUI performance overview with congestion handling.

Configuration steps for priority flow control on Cisco 3232C L3 switch

To ensure adequate bandwidth of shared ISL, determine the node-to-node traffic bandwidth and CoS value based on the business requirements.

Consider two factors while determining the bandwidth percentage for node-to-node traffic. The first factor is the bare minimum bandwidth required to facilitate node-to-node heartbeat and peak WRITE workload of host-to-node traffic. In FlashSystem, consider the node-to-node heartbeat at a maximum of 20 MBPS. If the peak WRITE workload for host-to-node traffic is 597 MBPS of the 10 Gbps (1250 MBPS) ISL, the node-to-node traffic will be 617 MBPS. Therefore, guarantee at least 50% of the total shared ISL bandwidth for node-to-node traffic.

To configure the switching settings for host-to-node and node-to-node traffic, the following sample settings can be used: allocate 10% bandwidth for each traffic type. These settings utilize priority flow control with CoS value 3 for node-to-node traffic and value 4 for host-to-node traffic.

	host-to-node traffic	node-to-node traffic
Representation object	CM_HOST_ATTACH_COS_4	CM_RDMA_CLUSTERING_COS_3
CoS value	4	3
% Of bandwidth allocation	10	10

Table 5: Bandwidth allocation and CoS values used.

1. Enable DCBx in the switch.

2. Reallocate TCAM space to enable traffic shaping and ensure sufficient buffer allocation, using the following commands.
```
switch(config-if)# hardware access-list tcam region racl-lite 512
switch(config-if)# hardware access-list tcam region qos-intra-lite
512
```

3. Create a class-map to classify node-to-node and host-to-node traffic by matching CoS value 4 and 3 respectively, using the following commands.
```
switch# configure
switch(config-if)# class-map type qos match-any CM_HOST_ATTACH_COS_4
switch(config-cmap-qos)# match cos 4
switch(config-if)# class-map type qos match-any
CM_RDMA_CLUSTERING_COS_3
switch(config-cmap-qos)# match cos 3
```

4. Define a policy-map of type QoS, configure two classes for each traffic type and set a QoS group with the desired CoS value by using the following commands.
```
switch(config-if)# policy-map type qos PM_TRAFFIC_MARKING
switch(config-pmap-qos)# class CM_HOST_ATTACH_COS_4
switch(config-pmap-c-qos)# set qos-group 4
switch(config-pmap-qos)# class CM_RDMA_CLUSTERING_COS_3
switch(config-pmap-c-qos)# set qos-group 3
```

5. Define a policy-map of type queuing and allocate bandwidth to 10 for both node-to-node and host-to-node traffic by using the following commands.
```
switch(config)# policy-map type queuing Storwize_ETH_Egress
switch(config-pmap-que)# class type queuing c-out-8q-q7
switch(config-pmap-c-que)# bandwidth percent 0
switch(config-pmap-que)# class type queuing c-out-8q-q6
switch(config-pmap-c-que)# bandwidth percent 0
switch(config-pmap-que)# class type queuing c-out-8q-q5
switch(config-pmap-c-que)# bandwidth percent 0
switch(config-pmap-que)# class type queuing c-out-8q-q4
switch(config-pmap-c-que)# bandwidth percent 10
switch(config-pmap-que)# class type queuing c-out-8q-q3
switch(config-pmap-c-que)# bandwidth percent 10
switch(config-pmap-que)# class type queuing c-out-8q-q2
```

```
switch(config-pmap-c-que)# bandwidth percent 0
switch(config-pmap-que)# class type queuing c-out-8q-q1
switch(config-pmap-c-que)# bandwidth percent 0
switch(config-pmap-que)# class type queuing c-out-8q-q-default
switch(config-pmap-c-que)# bandwidth remaining percent 100
```

6. Define a policy-map associated with policy type Network QoS and define classes for respective traffic, set the MTU and configure no-drop, by using the following commands. Frames with specified CoS values cannot be dropped.

```
switch(config)# policy-map type network-qos my8q-nq
switch(config-pmap-nqos)# class type network-qos c-8q-nq7
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos)# class type network-qos c-8q-nq6
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos)# class type network-qos c-8q-nq5
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos)# class type network-qos c-8q-nq4
switch(config-pmap-nqos-c)# pause pfc-cos 4
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos)# class type network-qos c-8q-nq3
switch(config-pmap-nqos-c)# pause pfc-cos 3
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos)# class type network-qos c-8q-nq2
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos)# class type network-qos c-8q-nq1
switch(config-pmap-nqos-c)# mtu 1500
```

7. Apply the defined policy map to the ingress and egress interface by using the following commands.

```
switch(config-if)# service-policy type network-qos my8q-nq
switch(config-if)# service-policy type queuing output
Storwize_ETH_Egress
```

8. Apply the following settings on each switch port connected to either IBM FlashSystem node or host port.

```
switch(config-if)# switchport
switch(config-if)# switchport trunk allowed vlan X*
switch(config-if)# priority-flow-control mode on
switch(config-if)# beacon
switch(config-if)# service-policy type qos input PM_TRAFFIC_MARKING
*X can be VLAN C or VLAN H based on the port type, Replace the value
of X with correct value
```

9. Perform the following settings on ISL ports.
Ensure that this port is exclusively being used as a bridge between the two switches on each site.

```
switch(config)# interface Ethernet1/32/1
switch(config-if)# switchport mode trunk
switch(config-if)# switchport trunk allowed vlan 1,C,H <<-- C & H
should be replaced by VLANs used for host-to-node & node-to-node
switch(config-if)# beacon
```

Configuring IBM FlashSystem

To configure IBM HyperSwap® solution over RDMA based Ethernet clustering, follow these steps.

1. Install RDMA capable adapter(s) that support identical technology (iWARP) on identical slots of each node.
2. Install SVC 8.5.2.0 or a higher code version on all FlashSystem nodes in the system.
3. Verify that the installed adapters with their respective technology are detected and shown up in the ‘sainfo lshardware’ CLI view.

```
panel_name 01-1
node_id 1
node_name node1
node_status Active
hardware 600
actual_different no
actual_valid yes
memory_configured 32
memory_actual 32
memory_valid yes
cpu_count 1
cpu_socket 1
cpu_configured 10 core Intel(R) Xeon(R) CPU E5-2618L v4 @ 2.20GHz
cpu_actual 10 core Intel(R) Xeon(R) CPU E5-2618L v4 @ 2.20GHz
cpu_valid yes
cpu_socket
cpu_configured
cpu_actual
cpu_valid
adapter_count 6
adapter_location 0
adapter_configured 12Gb/s SAS adapter
adapter_actual 12Gb/s SAS adapter
adapter_valid yes
adapter_location 0
adapter_configured Midplane bus adapter
adapter_actual Midplane bus adapter
adapter_valid yes
adapter_location 0
adapter_configured Four port 1Gb/s Ethernet adapter
adapter_actual Four port 1Gb/s Ethernet adapter
adapter_valid yes
adapter_location 1
adapter_configured Compression pass-through
adapter_actual Compression pass-through
adapter_valid yes
adapter_location 2
adapter_configured Two port 25Gb/s Ethernet iWARP adapter
adapter_actual Two port 25Gb/s Ethernet iWARP adapter
adapter_valid yes
adapter_location 3
adapter_configured Two port 25Gb/s Ethernet iWARP adapter
adapter_actual Two port 25Gb/s Ethernet iWARP adapter
adapter_valid yes
ports_different no
```

Figure 8: Example of ‘sainfo lshardware’ CLI.

4. Verify that all the ethernet ports are listed under ‘sainfo lsportip’ CLI view.
5. Assign IPv4 addresses to the adapter ports on all nodes in the system to establish node-to-node links using ‘satask chnodeip’ command.

IPv4 addresses on each node for identical port id must be on same subnets (i.e identical masks). Since these are on the same Ethernet domain specified gateway, they are ignored.

```
# satask chnodeip -ip 192.168.X.X -gw 192.168.X.1 -mask
255.255.255.0 -port_id 5 -vlan C
```

6. Configure the VLAN for node-to-node communication to isolate the network traffic by using the following command.

```
# satask chnodeip -ip 192.168.X.X -gw 192.168.X.1 -mask
255.255.255.0 -vlan C -port_id 5
```

where C can be any id configured on a switch. To establish a single clustering connection, both end points must use the same VLAN ID. However, if multiple clustering connections are required, each connection must have the same VLAN ID. In the case of two distinct connections, they can have different VLAN IDs.

VLAN must be configured on switches first and then on FlashSystem Nodes.

IPs can be listed using 'sainfo lsnodeip' CLI command.

```
IBM_FlashSystem:fab3_cl:superuser>sainfo lsnodeip
port_id    rdma_type port_speed vlan link_state state       node_IP_address gateway      subnet_mask
1                    1Gb/s          active     unconfigured
2                                   inactive   unconfigured
3                                   inactive   unconfigured
4                                   inactive   unconfigured
5          iWARP     25Gb/s     199  active     configured   192.168.12.15   192.168.12.1 255.255.255.0
6          iWARP     25Gb/s     199  active     configured   192.168.13.15   192.168.13.1 255.255.255.0
7          RoCE      25Gb/s          active     unconfigured
8          RoCE      25Gb/s          active     unconfigured
IBM_FlashSystem:fab3_cl:superuser>
```

Figure 9: Example of 'sainfo lsnodeip' CLI.

7. Verify the connectivity among the nodes after assigning IPs using 'sainfo lsnodeipconnectivity' CLI.

```
IBM_FlashSystem:fab3_cl:superuser> sainfo lsnodeipconnectivity
status         local_port_id local_vlan local_rdma_type local_ip_addr remote_port_id remote_vlan remote_rdma_type
remote_ip_addr remote_wwnn      remote_panel_name cluster_id       error_data
Connected:iWARP 5             199        iWARP          192.168.12.15 9             199         iWARP
192.168.12.13  50050768100032CA 02-2             0000020429A02218
Connected:iWARP 6             199        iWARP          192.168.13.15 10            199         iWARP
192.168.13.13  50050768100032CA 02-2             0000020429A02218
Connected:iWARP 5             199        iWARP          192.168.12.15 9             199         iWARP
192.168.12.16  5005076810001143 01-2             0000020429A02218
Connected:iWARP 6             199        iWARP          192.168.13.15 10            199         iWARP
192.168.13.16  5005076810001143 01-2             0000020429A02218
Connected:iWARP 5             199        iWARP          192.168.12.15 9             199         iWARP
192.168.12.14  500507681000ADE6 02-1             0000020429A02218
Connected:iWARP 6             199        iWARP          192.168.13.15 10            199         iWARP
192.168.13.14  500507681000ADE6 02-1             0000020429A02218
IBM_FlashSystem:fab3_cl:superuser>
```

Figure 10: Example of 'sainfo lsnodeipconnectivity' CLI.

Ensure that all ports are connected, recheck IP assignments and network settings to fix connectivity.

8. Create a system by using the 'satask mkcluster' command or configure the system using the GUI, after all nodes in the system are able to communicate with each other.
9. Set CoS value 3 for node-to-node traffic and value 4 for host-to-node traffic, using the following command.
```
# svctask chsystemethernet -hostattachcos 4 -systemcos 3
```

   Reinitiate the clustering session by performing an orderly warm start of all nodes except the config node to make the ‘`systemcos`’ settings effective.

   Note: Orderly warm start refers to the process of sequentially warm-starting clustered nodes, where the next iteration of warm start will wait for the last warm-started node to rejoin the cluster and ensure cluster stability.
10. Follow standard site configuration process using the following command.
```
# svctask chnode -site <site id 1 or 2> <node id>
```

    Refer to HyperSwap system configuration details on ibm.com for more information on configuration details.

11. Change topology of the system to HyperSwap using the following command.
```
# svctask chsystem -topology hyperswap
```

12. Configure IPs for host-to-node using ‘svctask mkip’ CLI command.
13. Create storage pool or mdiskgrp from the backend devices or drives such as mdisks or arrays, for each site separately.
    Storage pool or mdiskgrp can be created using the drives of available arrays with control enclosure or using mdisks mapped from iSCSI / FC backend.
14. Create HyperSwap volumes using the following command.
```
# svctask mkvolume -size <size of volume> -unit <gb, mb> -pool
<iogrp from site1:iogrp from site2>
```

15. Create a host using the following command.
```
# svctask mkhost -iscsiname <iSER/iSCSI initiator iqn>
```

16. Assign site to the host using the following command.
```
# svctask chhost -site <site id>
```

17. Map HyperSwap® volumes to hosts using the following command.
```
# svctask mkvdiskhostmap -host <host object id> <vdisk id>
```

18. After ‘svcinfo lshost’ CLI lists the host as offline due to a known issue, IO operations can be run from the host.

Summary

The paper focuses on the deployment of IBM FlashSystem as an IBM HyperSwap function in an Ethernet environment. It highlights the use of "Shared Inter-Switch Link (ISL)" in IBM HyperSwap configuration and provides a comprehensive guide on the complete configuration process, including details on the solution topology and offers insights into its cost-effectiveness and manageability.

Get more information

- Refer to IBM Spectrum Virtualize inter-node communication support over Ethernet-based RDMA for configuring RDMA based clustering setup using GUI.
- Refer to Configuring a system to use RDMA on ibm.com for RDMA based Ethernet clustering.
- Refer to HyperSwap function on ibm.com for more information on HyperSwap.
- Refer to Planning network cable connections on ibm.com for more details on supported iWARP adapter, cables, and SFP's.

About the authors

Shrirang Bhagwat is a lead developer in the Spectrum Virtualize team at IBM Systems Labs, India. His area of expertise is Ethernet-based block storage devices and has led development of the IP Replication feature. You can reach Shrirang at shbhagwa@in.ibm.com.

Abhishek Jaiswal is development lead in the Ethernet area for IBM Flash System product with expertise on High availability, resiliency, and host attachment areas such Clustering, Synchronous/Asynchronous Replication services, etc. You can reach Abhishek at ajaiswa9@in.ibm.com.

Aakanksha Mathur leads the test effort for IBM Spectrum Virtualize product chain at IBM Systems Labs, India. She has been working in the storage domain for more than 13 years including 10 years with IBM Systems Labs with expertise in block storage and now is actively involved in Ethernet enhancements in block storage. She can be reached at aamathur@in.ibm.com.

Hencilla Dsouza is a test engineer at IBM Systems Development Lab, working on Ethernet mission for Spectrum Virtualize. She can be reached at hencilla.dsouza@ibm.com.

IBM Corporation New Orchard Road Armonk, NY 10504

Produced in the
United States of America
May 2023