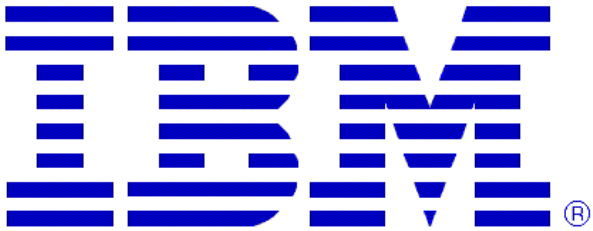

FICON Dynamic Routing (FIDR)

Technology and Performance Implications

August 2016

Pasquale “PJ” Catalano

Dr. Steve Guendert

Document WP102651

Systems Group

Notices, Disclaimer and Trademarks

Table of Contents

FICON Dynamic Routing (FIDR): Technology and Performance Implications

Executive Summary:

As part of the z13 announcement in January 2015, IBM announced support for a new FICON routing technique for FICON interswitch links (ISLs) called FICON Dynamic Routing (FIDR). FIDR enables ISL routes to be dynamically changed based on the Fibre Channel exchange ID, which is unique for each I/O operation. With FIDR, an ISL is assigned at I/O request time, so different I/Os from the same source port going to the same destination port may be assigned different ISLs.

z13 servers using FIDR have advantages for performance and management in configurations with ISL and cascaded FICON directors:

- Support sharing of ISLs between FICON and FCP (Metro Mirror/PPRC or distributed)
- I/O traffic is better balanced between all available ISLs
- Improve utilization of FICON directors and ISLs
- Easier to manage with a predictable and repeatable I/O performance

FICON dynamic routing can be enabled by definition of dynamic routing capable switches and control units in HCD. Also, z/OS has implemented a health check function for FICON dynamic routing.

This white paper will discuss requirements for FIDR, review the basics of cascaded FICON, explain how the predecessor technology (static routing) works, and compare it with the new FICON Dynamic Routing mechanism. The paper will conclude with a discussion of design considerations and the possible use cases where implementing FICON Dynamic Routing will be of the most benefit to z Systems end users.

Introduction

IBM originally announced FICON channels for the S/390 9672 G5 processor in May of 1998. Over the ensuing 18 years, FICON has rapidly evolved from 1 Gbit FICON Bridge Mode (FCV) to the 16 Gbit FICON Express16S channels that were announced in January 2015 as part of the IBM z Systems z13 announcement. While the eight generations of FICON channels have each offered increased performance and capacity, from the perspective of this paper the most important enhancements relate to the fibre channel upper level protocol (ULP) employed by FICON. Specifically, the first FICON ULP was called FC-SB-2. This specification supported native FICON storage devices, channel-to-channel connections, FICON directors (switches), as well as a number of other key features. However, the FC-SB-2 ULP did not support switched paths over multiple directors. When IBM introduced the FC-SB-3 ULP in January of 2003, this limitation was addressed. While FC-SB-2 employed a single byte link address that specified just the switch port on the director, FC-SB-3 employs a 2 byte address where the first byte is the address of the destination director and the second byte is the address of the port on the destination director. This is known as cascaded FICON. Since 2003, there have been many changes with how the FICON switching devices send fibre channel frame traffic between each other. The focus of this paper is the significant changes that have taken place in the past several years, with a particular emphasis on the most recent changes announced by IBM in January 2015.

Cascaded FICON Defined

Cascaded FICON refers to an implementation of FICON that allows two storage fabrics to be linked via connections between pairs of directors. The director-to-director connections are known as interswitch links (ISLs). ISLs support processor-to-processor, processor-to-disk or tape subsystem, and subsystem-to-subsystem logical switched connections. Cascaded FICON directors facilitate the design and implementation of robust disaster recovery/business continuity solutions such as IBM's Geographically Dispersed Parallel Sysplex™ (IBM GDPS®). Cascaded FICON architectures substantially reduce the infrastructure costs and complexities associated with these implementations. Cascaded directors also permit greater flexibility in the FICON architecture, more effective utilization of fibre links, and higher data availability in the enterprise.

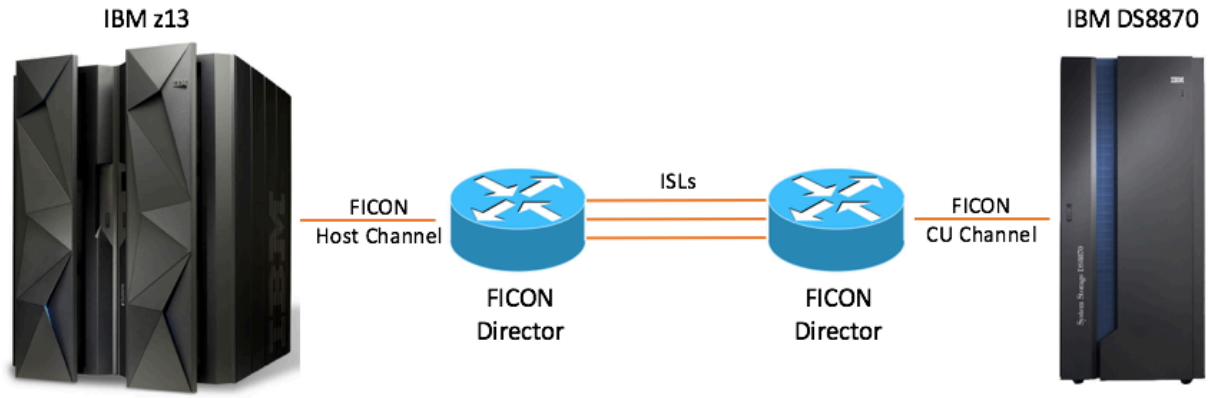

Figure 1: Cascaded Switch Configuration

As is shown in Figure 1, a logical connection over a cascaded FICON director configuration incorporates three end-to-end links. The first link connects the FICON channel N_Port (node port) to an F_Port (fabric port) on the FICON director. The second link connects an E_Port (expansion port) on the local director to an E_Port on the remote director. An inter-switch link (ISL) is a link between two switches, E_Port-to-E_Port. The ports of the two switches automatically come online as E_Ports once the login process finishes successfully. Finally, the third link connects the F_Port on the director to an N_Port on the control unit (CU) subsystem.

HCD defines the relationship between channels and directors (by switch ID) and specific switch port (as well as the destination switch ID for cascaded connections) for the switch to director segment of the link. However, HCD does not define the ISL connections. While this may initially appear to be surprising based on a reader's prior experience with the exacting specificity required by HCD, it means that the management of the traffic over ISLs is controlled exclusively by the directors. Hence, additional ISL bandwidth can be added to a configuration without any modification to the environment's HCD definitions. HCD simply assumes that the links are present and requires that 2 byte addressing be employed to define the connectivity to storage subsystems. The first byte is the switch ID and the second byte is the port to which the storage subsystem is connected. During initialization, the switches identify their peers and create a routing table so that frames for remote subsystems may be forwarded to the correct director.

Hardware Requirements for FIDR

FIDR has several hardware requirements that need to be met prior to implementation. There are z Systems, storage, SAN and network/DWDM device requirements.

Z Systems hardware requirements

FIDR requires z13/z13S running Driver 27L with bundle S08a or later with FICON Express 16S or FICON Express 8S channels.

DASD/Storage requirements

FIDR is supported with IBM DS8870 running firmware release 7.5 or later. FIDR will work with both 16Gbps and 8Gbps host adapter cards. Other storage OEMs may support FIDR as well. Clients should check with their DASD vendor to verify support.

SAN Hardware requirements

FIDR is supported on the IBM B type (Brocade) z Systems qualified Gen 5 products SAN768B-2 SAN384B-2, SAN 48B-5, and SAN42B-R (DCX 8510-8, DCX 8510-4, 6510, and 7840) and Gen 6 products (4 and 8 slot directors, G620 switch) running FOS levels 7.4.0a and later qualified versions. Brocade Network Advisor 12.4.1 or later qualified is also recommended.

FIDR is supported on the Cisco z Systems qualified products MDS 9710 and MDS 9706 running NxOS 6.2(11c), and MDS 9250i running NxOS 6.2(11d). Data Center Network Manager 7.2(1) or later qualified version is also recommended.

This document is not meant to determine qualified switch products. To ensure that the planned products to be implemented are qualified, registered users can see the IBM Resource Link® library for current information about qualified switch products and routing modes supported:

https://www-304.ibm.com/servers/resourcelink/lib03020.nsf/pages/switchesAndDirectorsQualifiedForIbmSystemZRFiconRAndFcpChannels?OpenDocument

Network/DWDM requirements

If a network/DWDM device is supported in a z Systems environment, in general it is supported for cascaded FICON and FIDR. A list of DWDM products qualified for use with z Systems can be found at IBM Resource Link® library:

https://www-304.ibm.com/servers/resourcelink/lib03020.nsf/pages/systemzQualifiedWdmProductsForGdpsSolutions?OpenDocument&pathID=

However, clients should also verify support of network/DWDM devices with their FICON SAN hardware components.

Software/OS requirements for FIDR

FIDR is supported on all operating systems that run on z13 and z13s that allow FICON channels to be defined.

Routing Overview and ISL Link Mapping

Data moves through a FICON SAN fabric from switch to switch and between storage devices and a mainframe along one or more paths that make up a route. Routing policies determine the path for each frame of data. Before the FICON SAN fabric can begin routing traffic, it must discover the route a frame should take to reach the intended destination. Route tables internal to the switches are listings that indicate the next hop to which a frame is directed to reach a destination.

The assignment of traffic between directors over the intervening ISLs is completely controlled by the directors. At the inception of FICON, the only mechanism available was the fabric shortest path first (FSPF) protocol. Fabric Shortest Path First (FSPF) is a link state selection protocol that directs traffic along the shortest path between the source and destination based upon the link cost. FSPF detects link failures, determines the shortest route for traffic, updates the routing table, provides fixed routing paths within a fabric, and maintains correct ordering of the frames. Once established, FSPF programs the hardware routing tables for all active ports on the switch.

Every time an ISL is added, the ISL traffic assignments change. While very simple, this technique is not attractive to an experienced mainframe architect since they cannot prescribe how the paths to a subsystem are mapped to the ISLs. In the worst case, all of the paths to a subsystem might be mapped to the same ISL. Moreover, once a path is assigned an ISL, that assignment is persistent.

FSPF is the foundation for all subsequent ISL routing mechanisms that have been introduced since 2003. Therefore, it would be beneficial to have a solid understanding of how FSPF works. The next section of this paper will review the FSPF protocol.

Review of Fabric Shortest Path First (FSPF)

The Fabric Shortest Path First (FSPF) protocol is the standardized routing protocol for Fibre Channel (hence FICON) SAN fabrics. It is the foundation for all of the routing mechanisms that will be discussed later in this paper; therefore, it is important to understand how FSPF works.

FSPF is a link state path selection protocol that directs traffic along the shortest path between the source and destination based upon the link cost. FSPF detects link failures, determines the shortest route for traffic, updates the routing table, provides fixed routing paths within a fabric, and maintains correct ordering of frames. FSPF also keeps track of the state of the links on all switches in the fabric and associates a cost with each link. The protocol computes paths from a switch to all the other switches in the fabric by adding the cost of all links traversed by the path, and chooses the path that minimizes the costs. This collection of the link states, including costs, of all the switches in the fabric constitutes the topology database or link state database.

FSPF is based on a replicated topology database that is present in every switching device in the FICON SAN fabric. Each switching device uses information in this database to compute paths to its peers via a process known as path selection. The FSPF protocol itself provides the mechanisms to create and maintain this replicated topology database. When the FICON SAN fabric is first initialized, the topology database is created in all operational switches. If a new switching device is added to the fabric, or the state of an interswitch link (ISL) changes, the topology database is updated in all the fabric's switching devices to reflect the new configuration.

A Link State Record (LSR) describes the connectivity of a switch within the topology database. The topology database contains one LSR for each switch in the FICON SAN fabric. Each LSR consists of a link state record header, and one or more link descriptors. Each link descriptor describes an ISL associated with that switch. A link descriptor identifies an ISL by the Domain_ID and output port index of the "owning" switch and the Domain_ID and input port index of the "neighbor" switch. This combination uniquely identifies an ISL within the fabric. LSRs are transmitted during fabric configuration to synchronize the topology databases in the attached switches. They are also transmitted when the state of a link changes and on a periodic basis to refresh the topology database.

Associated with each ISL is a value known as the link cost, which reflects the desirability of routing frames via that ISL. The link cost is inversely proportional to the speed of the link-higher speed links are more desirable transit paths, therefore they have a lower cost. The topology database has entries for all ISLs in the fabric, enabling a switch to compute its least cost path to every other switching device in the FIOCN SAN fabrics from the information contained in its copy of the database.

There are three main functions associated with the FSPF protocol:

1) A Hello protocol to establish two way communication with a neighbor switch and determine the connectivity of an ISL.
2) An initial topology database synchronization protocol.
3) A topology database maintenance protocol.

The FSPF Hello Protocol

After a FICON switching device acquires a Domain_ID, it may begin the process of building a routing table in order to begin delivering frames. The Hello protocol is used to determine when a neighbor switch is ready to begin routing frames on an ISL. Once a switch acquires a Domain_ID, it begins transmitting periodic Hello messages, and indicates the source of the ISL by using the Domain_ID and output port index of the sending switching device. Initially, a FICON switching device does not know the Domain_ID of its neighbor(s) and the switch is then said to have "one-way" communications with the neighbor. When a FICON switching device receives a Hello on an ISL, it learns the Domain_ID of the neighbor, and its uses its output port on this ISL to send its own Hello containing its own Domain_ID and port index. The switch is then aware of the neighbor switch's Domain_ID and includes that in all subsequent Hellos sent on the ISL. If a switching device receives a Hello containing its Domain_ID as the recipient Domain_ID, the neighbor switch is said to know the Domain_ID of this switch, and the switch also knows the Domain_ID of the neighbor switch. At this point, the two are said to have "two-way" communications with each other and they can begin their initial topology database synchronization.

Hello messages are transmitted on a periodic basis on each ISL even after two-way communication is established. The periodic Hello transmission provides a mechanism to detect a switch or an ISL failure. In essence, the periodic Hello acts as a "heartbeat" between the switching devices. The time interval between successive Hellos (in seconds) is defined by a timer called the Hello interval. The number of seconds that a switching device will wait for a Hello before reverting back to one-way communication is defined by the Dead interval. Both the Hello and Dead intervals are communicated between switches as part of the Hello itself. If a switch fails to receive a Hello within the expected time it 1) assumes the neighbor switch is no longer operational and 2) removes the associated ISL from its topology database. At this time the switch removes the Domain_ID of the neighbor switch from the Hello messages and the

Hello protocol reverts to the on-way communication state. If a link failure occurs on an ISL, two-way communication is lost. When the ISL is restored, the switching devices must re-establish two-way communication and synchronize their topology databases before the ISL may be again used for routing frames.

Initial Topology Database Synchronization

Once two-way communication has been established via the Hello protocol, the switches synchronize their topology databases. During the initial topology database synchronization, each switch sends its entire topology database to its neighbor switch(es). When a switch completes sending its database, it sends a Link Status Update (LSU) with no LSRs. When it receives an acknowledgement to this LSU, topology database synchronization is complete, and the switches are then said to be "adjacent" on that ISL. The ISL is now in the "full state" and may be used for frame delivery.

While the entire topology database is exchanged during this initial synchronization process, LSRs are also transmitted during the database maintenance phase to reflect topology changes in the FICON SAN. The topology database must be updated whenever the state of any ISL changes. The specific events that will cause such an LSR to be transmitted include:

1) An ISL fails (or the entire switch connected to the ISL fails). A new LSR is transmitted to remove the failed ISL(s) from the topology database.
2) An ISL reverts to one-way communication status. A new LSR is transmitted to remove the one-way ISL from the topology database.
3) A new ISL completes the link initialization and the initial database synchronization. An LSR is transmitted to notify the other switches in the fabric to add the new information to their databases.

Path Selection

Once the topology database has been created and a switch has information about available paths, it can compute its routing table and select the paths that will be used for forward frames. Fibre Channel uses a least cost approach to determine paths used for routing frames. Each ISL is assigned a link cost metric that reflects the desirability of using that link. The cost metric is based on the speed of the ISL and an administratively applied cost factor. The link cost us calculated using the following formula:

$$\text{Link Cost} = S \times (1.0625e12 / \text{Baud Rate})$$

The factor S is an administratively defined factor that may be used to change the cost of specific links. By default, S is set to 1. Applying the formula, assuming S is set to 1, the link cost for a 1Gbps links is 1000, while a 2Gbps ISL cost=500.

When there are multiple paths with the same cost to the same destination (which is typical for 99.9% of cascaded FICON SAN architectures), a switching device must decide how to use these paths. It may select one path only (not ideal) or it may attempt to balance the traffic among the available paths to avoid congestion of ISLs. If a path fails, the switch may select an alternate ISL for frame delivery. This is where the different types of ISL routing come into play. These routing techniques are the subject of the next section.

FICON Routing Mechanisms

As was previously discussed, the assignment of traffic between directors over the intervening ISLs is completely controlled by the directors. At the inception of cascaded FICON, the only mechanism available for this was fabric shortest path first (FSPF). Every time an ISL is added, the ISL traffic assignments change. While very simple, this technique is not attractive to an experienced mainframe architect since they cannot prescribe how the paths to a subsystem are mapped to the ISLs. In the worst case, all of the paths to a subsystem might be mapped to the same ISL. Moreover, once a path is assigned an ISL, that assignment is persistent. Over the past several years, multiple techniques beyond simple FSPF have been introduced for routing traffic on FICON ISLs. These techniques fall under two categories: static routing, and dynamic routing.

Traditional static routing mechanisms

In static routing the choice of routing path is based only on the incoming port and the destination domain.Brocade Port Based Routing (PBR) assigns the interswitch link (ISL) statically based on "first come first served" at fabric login (FLOGI). The switch has no idea whether the port that is logging in will use the ISL and whether it will cause bottleneck. The actual data workload is not balanced across the ISLs. This can result in some ISLs being overloaded, and some being underutilized. Brocade Device Based Routing (DBR) and Cisco's Source/Destination (SID/DID) Routing optimizes routing path selection and utilization based on a hash of the Source ID (SID) and Destination ID (DID) of the path source and destination ports. Therefore, the ingress ports that are only passing traffic locally and not using the ISLs do not get assigned to an ISL. As a result, every distinct flow in the fabric can take a different path through the fabric. This helps to ensure that the exchanges between a pair of devices stay in order. The effect of this is better workload balancing across ISLs. Finally, the routing table changes after every power on reset (POR) on a z Systems server, so the ISL assignment is somewhat unpredictable.

Static routing drawbacks

The static routing policies/mechanisms that have been required until the GA of FIDR posed some problems based on how they worked. PBR assigned the ISL routes to the ingress ports at Fabric Login (FLOGI) and it was done on a first come first served basis. Which ISL was actually assigned was done on what was termed a round-robin basis. While ingress port assignments would be balanced, the actual data traffic that would flow across the ISLs would often times not be balanced. In other words, static routing policies may not exploit all the available ISL bandwidth.

The path selection algorithms used by the IBM synchronous replication technology, Metro Mirror (previously known as Peer-to-Peer Remote Copy or PPRC), is based on the link bandwidth. The IBM DS8000 series DASD array keeps track of outstanding Metro Mirror I/O operations on each port. DS8000 path selection chooses the next port to schedule work based on the link with the most available bandwidth. Thus, Metro Mirror traffic should also steer around SAN congestion towards better performing routes. Consider a configuration with four Metro Mirror links from the primary control unit to the secondary control unit. These four Metro Mirror ports go to a fabric with eight ISLs. At most, four ISLs will be used. The hashing algorithm might actually map traffic of two or more of the PPRC ports onto the same ISL resulting in fewer than four ISLs actually being used. In many cases, static routing policies will do an adequate job spreading the work across many routes. However, in others, exploitation of the link bandwidth can be suboptimal.

For another example, in a scenario with 12 ingress ports and 4 ISLs, PBR would assign 3 ingress ports to each ISL in a round robin fashion. Ingress ports were assigned to ISLs regardless of whether or not they would be sending traffic across the ISLs. This is why it was very common to see PBR implementations result in 50% or more of the total cross site data workloads being on one ISL, and perhaps no data at all on other ISLs. For large z/OS Global Mirror implementations, this kind of behavior by PBR would result in many suspends and very poor replication performance.

DBR improved things somewhat by making the routing assignments based on source ID/destination ID (SID/DID), so the assignment of ingress ports to ISLs actually did require the ingress ports to be used for sending traffic across the ISLs. However, the other primary drawback of static routing policies still applied: routing can change each time the switch is initialized which leads to unpredictable, non-repeatable results.

However, static routing policies do have some advantages. One outstanding example is provided with how the routing policies handle the impact of a slow drain device. Static routing has the advantage of limiting the impact of a slow drain device or a congested ISL to a small set of ports that are mapped to that specific ISL. If congestion occurs in an ISL, the z Systems channel path selection algorithm will detect the congestion through the increasing initial command response (CMR) time in the in-band FICON measurement data. The z Systems channel subsystem begins to steer the I/O traffic away from congested paths and toward better performing paths using the CMR time as a guide. Additional host recovery actions are also available for slow drain devices.

IBM z13 and FICON Dynamic Routing (FIDR)

Until z13, static routing polices were required to guarantee in-order processing. With the IBM z13, IBM announced that FICON channels will no longer be restricted to the use of static Storage Area Network (SAN) routing policies. The z Systems servers now support dynamic routing in the SAN with the FICON Dynamic Routing (FIDR) feature. In other words, end users with cascaded FICON director architectures will no longer have to rely on static routing polices such as Port Based Routing (PBR) or Device Based Routing (DBR). A dynamic routing policy means that there are no fixed routes from the source ports to the destination ports. FIDR is designed to support the dynamic routing policies provided by the FICON director manufacturers.

Routes across ISLs are assigned to I/O operations dynamically on a per exchange (per I/O) basis. Loading of ISLs is applied at the time of the data flow. This provides the most effective mechanism for balancing data workload traversing available ISLs. Thus, every exchange can take a different path through the fabric. There are no longer fixed routes from the source ports to the destination ports. ISL resources are utilized equally and therefore, the ISLs can run at higher utilization rates without incurring queuing delays. High workload spikes resulting from peak period usage and/or link failures can also be dealt with more easily with FIDR.

One example of such a dynamic routing policy is Brocade's Exchange Based Routing (EBR). The second example is Cisco's originator exchange ID (OxID) routing. It should be noted that routing policies also impact local switching as well as ISLs. Clients should check with their FICON switch vendor for their specific support statement.

FICON Dynamic Routing Explained

With FICON Dynamic Routing (FIDR) the routing assignments are based on the SID/DID and the Fibre Channel originator exchange ID (OXID). Routes across ISLs are assigned to I/O operations dynamically on a per exchange (per I/O) basis. Loading of ISLs is applied at the time of the data flow. This provides the most effective mechanism for balancing data workload traversing available ISLs. Thus, every exchange can take a different path through the fabric. There are no longer fixed routes from the source ports to the destination ports. ISL resources are utilized equally and therefore, the ISLs can run at higher utilization rates without incurring queuing delays. High workload spikes resulting from peak period usage and/or link failures can also be dealt with more easily with FIDR.

This improves utilization of all available paths, thus reducing possible congestion on the paths. Every time there is a change in the network which changes the available paths, the traffic can be redistributed across the available paths.

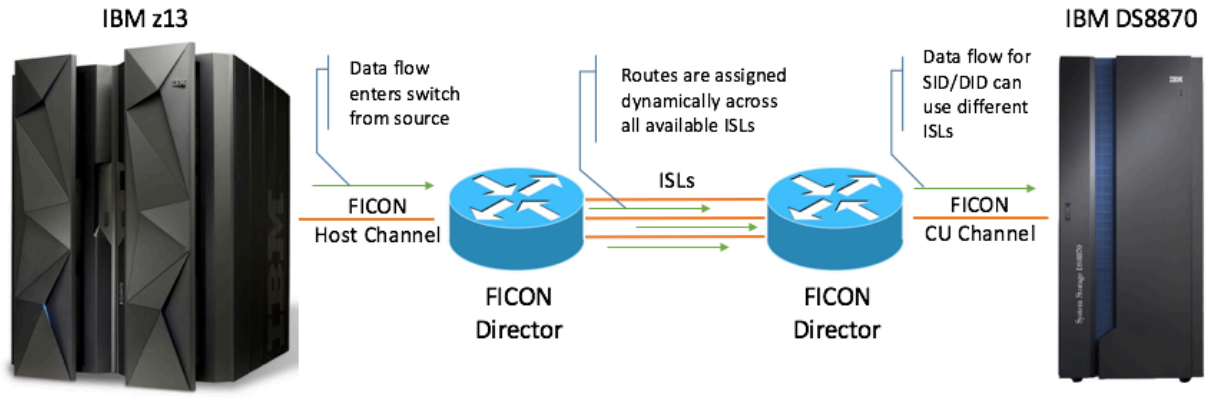

Figure 2: Details of FIDR

FIDR Benefits, Use Cases and Design Considerations

FIDR Benefits

The characteristics of FIDR provide two primary benefits to the end user. Which benefit applies most will depend on your specific circumstances and configuration.

1) Potential bandwidth consolidation/ISL consolidation. For many years the SAN vendor and IBM have recommended keeping FCP traffic (such as PPRC/Metro Mirror) and FICON traffic on their own ISLs or own group of ISLs-in other words segregate the traffic types. With FIDR, it's now possible to take advantage of FIDR and share ISLs among previously segregated traffic. This leads to the obvious consolidation cost saving in the hardware: fewer ISLs means fewer FICON director ports and/or DWDM links. However, that cost savings could be minuscule in comparison to the potential bandwidth cost savings. The potential bandwidth cost savings depends on an end users specific circumstances. In Asia and the Americas, the distance between data centers is usually greater than it is in Europe. Many large end users now own their bandwidth. Also, what type of bandwidth are we talking about? Those are all factors.

2) Better utilization of available bandwidth/assets. This is likely to be most attractive to PPRC/Metro Mirror users and/or end users who own their own bandwidth. Many such end users will prefer to keep their traffic types segregated and not share ISLs. One of the drawbacks to using static

routing with PPRC was that some ISLs might not be used at all-they would only be there for failover. Consider a configuration with four PPRC links from a primary control unit to the secondary control unit and these four PPRC ports connect to a fabric with eight ISLs. At most, with static routing only four of these ISLs would be used and the other four would be there for redundancy/failover. With FIDR, all eight ISLs will be used which is far more efficient utilization of the available bandwidth and could have a positive impact on performance.

Design Considerations

There are two behaviors that may occur in a FICON SAN with FIDR enabled that z Systems end users need to be aware of. The first behavior is dilution of error threshold counts. The second is the effects of slow drain devices.

The first behavior of concern is dilution of error threshold counts. It is possible for errors to occur in a FICON SAN that cause Fibre Channel frames to get lost. One example scenario is triggered when there are more than 11 bit errors in a 2112 bit block of data and Forward Error Correction (FEC) code cannot correct this large of a burst of errors. For z Systems and the FICON protocol, an error is then detected by the host operating system. This error will typically be an interface control check (IFCC) or a missing interrupt. The exact error type depends on exactly where in the I/O operation the error occurs. z/OS counts these errors and are tracked back to a specific FICON channel and control unit link. With static routes the error count covers the specific ISL that is in the path between the FICON channel and the control unit. With FICON dynamic routing mechanisms, the intermittent errors caused by a faulty ISL will occur on many channel to control unit links, unknown to the operating system or the z Systems processor itself. Therefore, the error threshold counters can get diluted: they get spread across the different operating system counters. This then makes it entirely possible that either the thresholds are not reached in a time period needed to recognize the faulty link, or that the host thresholds are reached by all control units that cross the ISL in question, resulting in all the channel paths being fenced by the host operating system. To prevent this behavior, the end user should use the FICON SAN switching devices capabilities to set tighter error thresholds internal to the switch and fence/decommission faulty ISLS before the operating system's recovery processes get invoked.

The second behavior to be aware of/concerned with when using FIDR is more important: slow drain devices and their impact on the FICON SAN. Recall that a slow drain device is a device that does not accept frames at the rate generated by the source. When slow drain devices occur, FICON SANs are likely to lack buffer credits at some point in the architecture. This lack of buffers can result in FICON switching device port buffer credit starvation, which in the extreme cases can result in congested or even completely choked ISLs. Frames that are "stuck" for long periods of time (typically for periods >500 milliseconds) may ultimately be dropped by the FICON SAN fabric, resulting in errors being detected. The most common type of error in this situation described would be C3 discards.

When a slow drain device event does occur, and corrective action is not taken in short order, ISL traffic can become congested as the effect of the slow drain device propagates back into the FICON SAN. With FICON dynamic routing policies being implemented a slow drain device may cause the buffer to buffer credit problem to manifest itself on all ISLs that can access the slow drain device. This congestion spreads and can potentially impact all traffic that needs to cross the shared pool of ISLs. With static routing policies, the congestion and its impact is limited to the one ISL that accesses the slow drain device.

Possible causes of slow drain devices include, but are not limited to:

1) An insufficient number of buffer credits configured for links that access devices a long distance away.

2) Disparate link speeds between the FICON channel and control unit links.

3) Significant differences in cable lengths.

4) Congestion at the control unit host adapters caused when the link traffic (across all ISLs) exceeds the capacity of the host adapter. This can occur when too many Metro Mirror ISLs share the same host adapter as FICON production traffic ISLs.

The z13 processor and z/OS have capabilities that mitigate the effect of slow drain devices, such as channel path selection. The algorithms steer the I/O workload away from the paths that are congested by the slow drain device towards the FICON channels in a separate, redundant FICON SAN. Best practices for z Systems I/O configurations require at least two separate and redundant FICON SANs. Many end users use four and the largest configurations often use eight.

For Metro Mirror traffic, best practices call for the z Systems end user to use FIDR in the fabric for predictable/repeatable performance, resilience against work load spikes and ISL failures, and for optimal performance. It is felt that if a slow drain device situation does occur in a FICON SAN fabric with Metro Mirror traffic, it will impact the synchronous write performance of the FICON traffic because the write operations will not complete until the data is synchronously copied to the secondary control unit. Since FICON traffic is already subject to the slow drain device scenarios today, exploiting FIDR does not introduce a new challenge to the end user and their FICON workloads.

Bandwidth Sizing Between Sites, Virtual Fabrics and DWDM Considerations

In any cascaded FICON architecture, it is a best practice to perform a bandwidth sizing study. Such a study can help you determine the bandwidth required for the cascaded links, the number of ISLs required to support that bandwidth requirement, as well as allow you to plan for anticipated storage growth and associated bandwidth growth over a multi-year period. Factors to consider in the bandwidth sizing study include:

1) What type of traffic is going across the cascaded FICON links (ISLs)? DASD, tape, CTC, all of the above?
2) Do I have replication traffic going across the ISLs? If so, what type (synchronous, asynchronous, both)?
3) Do I wish to isolate a particular traffic type to its own set of ISLs (by OS, storage type, replication type)?
4) Am I using trunking/port channels in conjunction with FIDR?
5) Am I using virtual fabrics and how does that effect my ISL allocation?
6) Do I have an SLA that must be met on a specific replication traffic?
7) Does my environment include a GDPS or similar architecture? Will I be performing a HyperSwap?

There are a variety of tools that can be used for this study. Some of the better tools include the Intellimagic family of software.

If you are employing a virtual fabrics architecture, it is recommended that you consult with your switch vendor for specific configuration guidelines on ISL usage with FIDR and virtual fabrics.

Finally, if you are employing DWDM devices in a cascaded FICON architecture, check with your switch vendor **AND** DWDM vendor for specifics on FIDR support, as well as support for other cascaded FICON technologies (for example, not every DWDM vendor supports VC_RDY and requires use of R_RDY on ISLs).

FIDR consistency reporting

The dynamic routing health check (IOS_DYNAMIC_ROUTING) is part of the z/OS 2.2.0 IOS checks in the IBM Health Checker for z/OS. This check identifies any inconsistencies in the dynamic routing support within the SAN. In order for dynamic routing to function properly, dynamic routing must be supported at all endpoints, the channel and connected devices, communicating through the switches. When dynamic routing is enabled in the SAN, the z/OS Health Check will verify that the processor and attached DASD, tape, and non-IBM devices defined as type CTC support dynamic routing and will identify those endpoints that do not.

The system runs this check whenever any of the following occur:

1) At IPL when IBM Health Checker for z/OS starts.
2) A change in the dynamic routing support within the SAN is detected
    a. CHPID configured online or offline
    b. Switch device varied online
    c. First device in a storage controller comes online
3) A user manually runs the health check

Clients must define a Control Unit Port (CUP) device to z/OS for each switch connected to each channel used by z/OS. z/OS uses the CUP device to gather information from the switch (topology, performance statistics for RMF, etc). New information is returned from the CUP that indicates whether dynamic routing is enabled for the SAN fabric. If the processor (channels) does not support dynamic routing, z/OS checks whether any channel is connected to a switch enabled for dynamic routing. Each control unit that is switch connected is checked to determine whether its channels are connected to a switch enabled for dynamic routing. The storage controller returns a dynamic routing capability flag in the self-description data. When VERBOSE=YES is specified on the check, the report will also include the processor or any controllers that would need to support dynamic routing if dynamic routing were enabled in the SAN. This helps plan for future enablement of dynamic routing in the SAN.

Figures 3 is an example of the output of this check where all devices and hosts support FIDR, an no inconsistencies are detected.

```
CHECK(IBMIOS,IOS_DYNAMIC_ROUTING)
SYSPLEX: ENGTEST1 SYSTEM: S01
START TIME: 08/02/2016 13:12:20.844419
CHECK DATE: 20150901 CHECK SEVERITY: MEDIUM

IOSHC144I Dynamic routing is enabled in the SAN and no
inconsistencies were detected.
```

Figure 3: Health Check Output of Supported Devices

Figure 4 is an example of the output where not all devices support FIDR. As you can see, FIDR inconsistencies were detected, and therefore should not be enabled for this fabric.

```
CHECK(IBMIOS,IOS_DYNAMIC_ROUTING)
SYSPLEX: ENGTEST1 SYSTEM: S01
START TIME: 08/02/2016 11:45:45.122744
CHECK DATE: 20150901 CHECK SEVERITY: MEDIUM

IOSHC144I Dynamic routing is enabled in the SAN but not supported by the
following controller(s):

   NODE DESCRIPTOR
   002107.932.IBM.75.0000000DT781
   002107.932.IBM.75.0000000Y4421
   002107.941.IBM.75.0000000PG761
   002107.951.IBM.75.0000000ZZ251
   Unknown ND for CU 8020
   Unknown ND for CU 8030

* Medium Severity Exception *

IOSHC142E Dynamic routing inconsistencies were detected
```

Figure 4: Health Check Output of Non-Supported Devices

Mixed Capability Devices and coexistence of policy types

Not every end user can simply purchase all new mainframes, storage, and FICON switching devices that will support FIDR. At the time of this writing, IBM intends on providing new z/OS functions to help end users manage the coexistence of FIDR capable devices and non-FIDR devices in the same architecture. For example, z Systems, z/OS, FICON SAN vendors, and storage vendors will all support a new SAN health check to detect when non-FIDR- capable devices and non-FIDR-capable processors are attached to switches running with dynamic routing policies enabled. This alerts clients to the possible I/O errors that can occur. Additionally, new configuration definition options are supported by the z Systems Hardware Configuration Definition (HCD) dialog to allow clients to designate which switches and devices are intended to be run with dynamic routing policies enabled. This will help prevent clients from accidentally dynamically adding a non-FIDR capable device to a FICON director running with FIDR enabled. So, co-existence of dynamic routing and static routing policies in the same FICON environment will be possible.

Summary

A great deal of technological innovation has occurred over the past twelve years since cascaded FICON was initially introduced by IBM in 2003. These technical innovations have driven the need for further innovations in how the interswitch links manage traffic flows, and how the end user manages the ISLs. FICON Dynamic Routing is the latest such management innovation. While it can have some potential points of concern, FIDR provides significant technical improvements over the older FICON static routing polices. FIDR also has the potential to allow the end user to realize significant cost savings in terms of architectural and bandwidth requirements. Given that the behaviors of concern, such as slow drain devices can be managed by the end user using good discipline, the Health Checker for z/OS and the FICON SAN vendor management tools, the benefits of implementing FIDR should far outweigh the potential issues.

Revision History

3/2018: Authors update.

