Understanding NPIV and the Performance of Channels with zLinux

Dr. Steve Guendert, Brocade Communications

Abstract

- This session will be two parts. The first part will discuss how mainframe I/O channels perform using QDIO in a zLinux environment. The functionality of QDIO will be explained and how the zLinux I/O works on the modern day System z. The second part of the presentation will discuss Node Port ID Virtualization (NPIV) and how a System z10 can make use of NPIV technology to virtualize its connectivity to storage, and realize TCO savings in the process.

Agenda

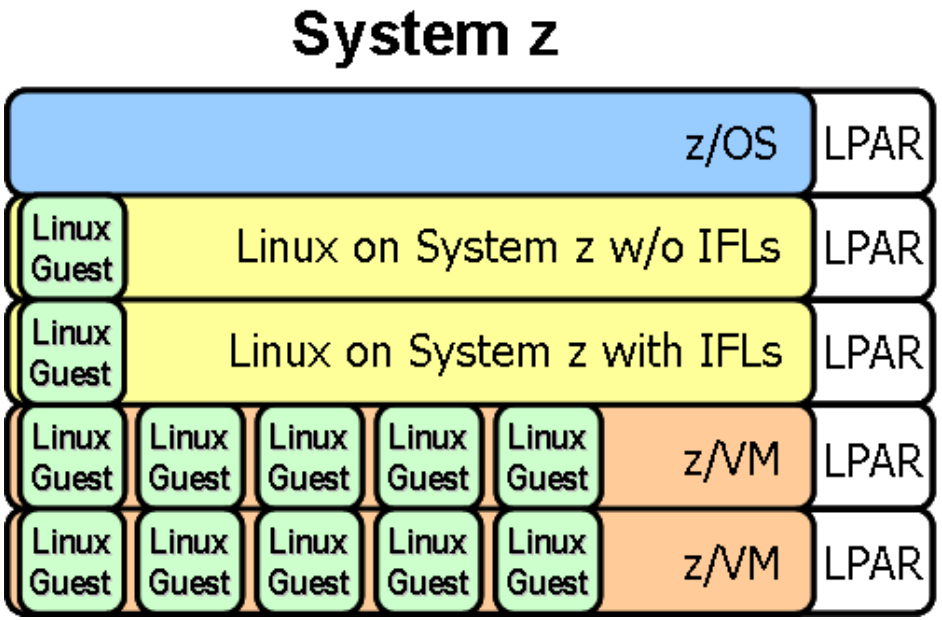
- Background/introduction
- FCP Channels on the mainframe
- FCP channel performance
- Challenges of access to distributed storage
- NPIV

Key References

- S. Guendert. *Understanding the Performance and Management Implications of FICON/FCP Protocol Intermix Mode (PIM).* Proceedings of the 2008 CMG. Dec 2008.
- I. Adlung, G. Banzhaf et al. "FCP For the IBM eServer zSeries Systems: Access To Distributed Storage". *IBM Journal of Research and Development.* 46 No.4/5, 487-502 (2002).
- J. Srikrishnan, S. Amann, et al. "Sharing FCP Adapters Through Virtualization." *IBM Journal of Research and Development.* 51 No. ½, 103-117 (2007).
- J. Entwistle. "IBM System z10 FICON Express8 FCP Channel Performance Report". IBM 2009.
- American National Standards Institute. "Information Technology-Fibre Channel Framing and Signaling (FC-FS)." ANSI INCITS 373-2003.
- G. Bahnzhaf, R. Friedrich, et al. "Host Based Access Control for zSeries FCP Channels", *z/Journal* 3 No.4, 99-103 (2005)
- S. Guendert. "The IBM System z9, FICON/FCP Intermix, and Node Port ID Virtualization (NPIV). *NASPA Technical Support.* July 2006, 13-16.
- G. Schulz. *Resilient Storage Networks.* pp78-83. Elsevier Digital Press. Burlington, MA 2004.
- S. Kipp, H. Johnson, and S. Guendert. "Consolidation Drives Virtualization in Storage Networks". *z/Journal.* December 2006, 40-44.
- S. Kipp. H. Johnson, and S. Guendert. "New Virtualization Techniques in Storage Networking: Fibre Channel Improves Utilization and Scalability." *z/Journal*, February 2007, 40-46

Background/Intro: Linux on System z

- Linux on System z was ten years old in 2009
- Virtualization is a key component to address IT's requirement to control costs yet meet business needs with flexible systems
- System z Integrated Facility for Linux (IFL) leverages existing assets and is dedicated to running Linux workloads while containing software costs
- Linux on System z allows you to leverage your highly available, reliable and scalable infrastructure along with all of the powerful mainframe capabilities
- Your Linux administrators now simply administer Linux on a "Big Server"



Background-Intro: Protocol Intermix

- FCP channel support introduced in May 2002
- What is PIM?
- Why PIM?

FCP channels on the mainframe

FICON and FCP Mode

- A FICON channel in Fibre Channel Protocol mode (CHPID type FCP) can access FCP devices through a single Fibre Channel switch or multiple switches to a SCSI device

- The FCP support enables z/VM, z/VSE, and Linux on System z to access industry-standard SCSI devices. For disk applications, these FCP storage devices use Fixed Block (512-byte) sectors instead of Extended Count Key Data (ECKD) format.

- FICON Express8, FICON Express4, FICON Express2, and FICON Express channels in FCP mode provide full fabric attachment of SCSI devices to the operating system images, using the Fibre Channel Protocol, and provide point-to-point attachment of SCSI devices.

FICON and FCP Mode (Continued)

- The FCP channel full fabric support enables switches and directors to be supported between the System z server and SCSI device, which means many “hops” through a storage area network (SAN).

- FICON channels in FCP mode use the Queued Direct Input/Output (QDIO) architecture for communication with the operating system.

FCP channels

- “Classic” zSeries operating systems such as z/OS and z/VM were designed for use only with storage controllers that support the I/O protocols defined by the z/Architecture.
- This changed with the advent of Linux for zSeries since its storage I/O component is oriented toward SCSI protocols.
- This necessitated adding specific support to Linux for zSeries to enable it to function in a CCW based zSeries I/O environment.

FCP channels and QDIO

- FICON channels in FCP mode use the Queued Direct Input/Output (QDIO) architecture for communication with the operating system.
- This is derived from the QDIO architecture initially defined for OSA Express and for Hipersockets communications.
- FCP channels do not use control devices.
- Instead, data devices representing QDIO queue pairs are defined:
  - Request queue
  - Response queue

QDIO queue pair basics

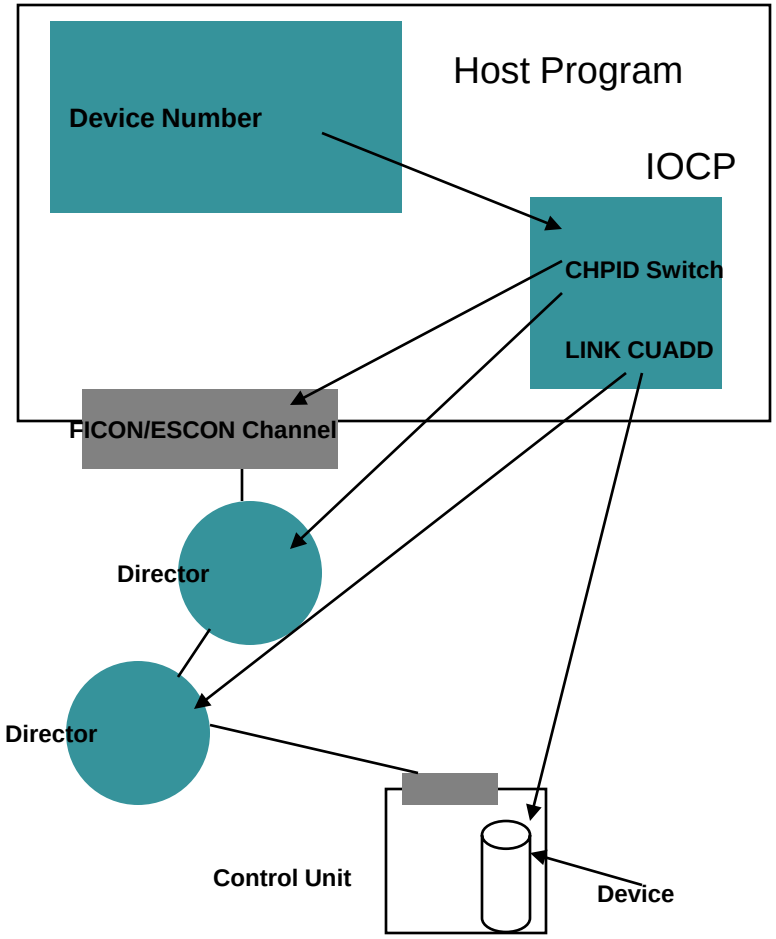
- Each of these QDIO queue pairs represent a communication path between an operating system and the FCP channel.
- FCP requests are sent by the operating system to the FCP channel via the request queue.
- Response queue is used to pass complete indications and unsolicited status indications to the operating system.
- Use of “Sense” CCWs for channel programs

QDIO and hardware definitions

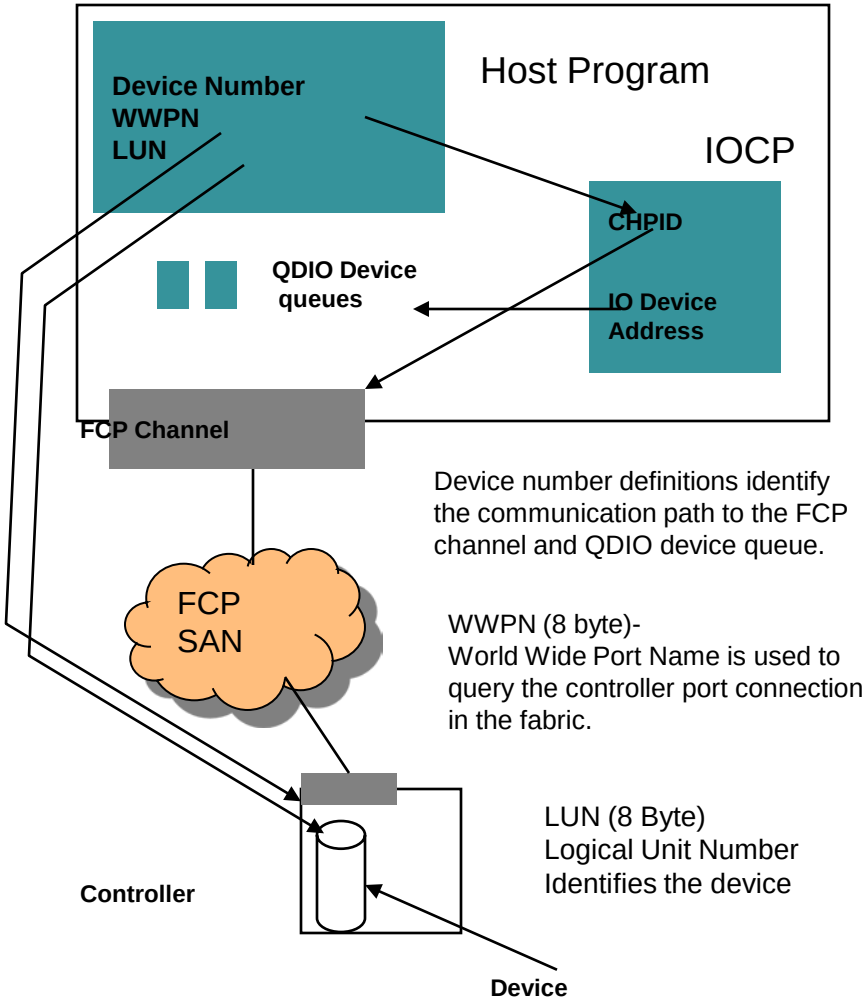
- HCD/IOCP is used to define the FCP channel type and QDIO data devices.
- There is no definition requirement for fibre channel storage controllers and devices, nor switches and directors.
- These devices are addressed using World Wide Names (WWNs), Fibre Channel Identifiers (IDs) and Logical Unit Numbers (LUNs).
  - These addresses are configured on an operating system level
  - They are passed to the FCP channel together with the corresponding Fibre Channel I/O or service request via a logical QDIO device (queue).

I/O definition comparison

Classical System z I/O definitions



System z FCP (SCSI) I/O definitions

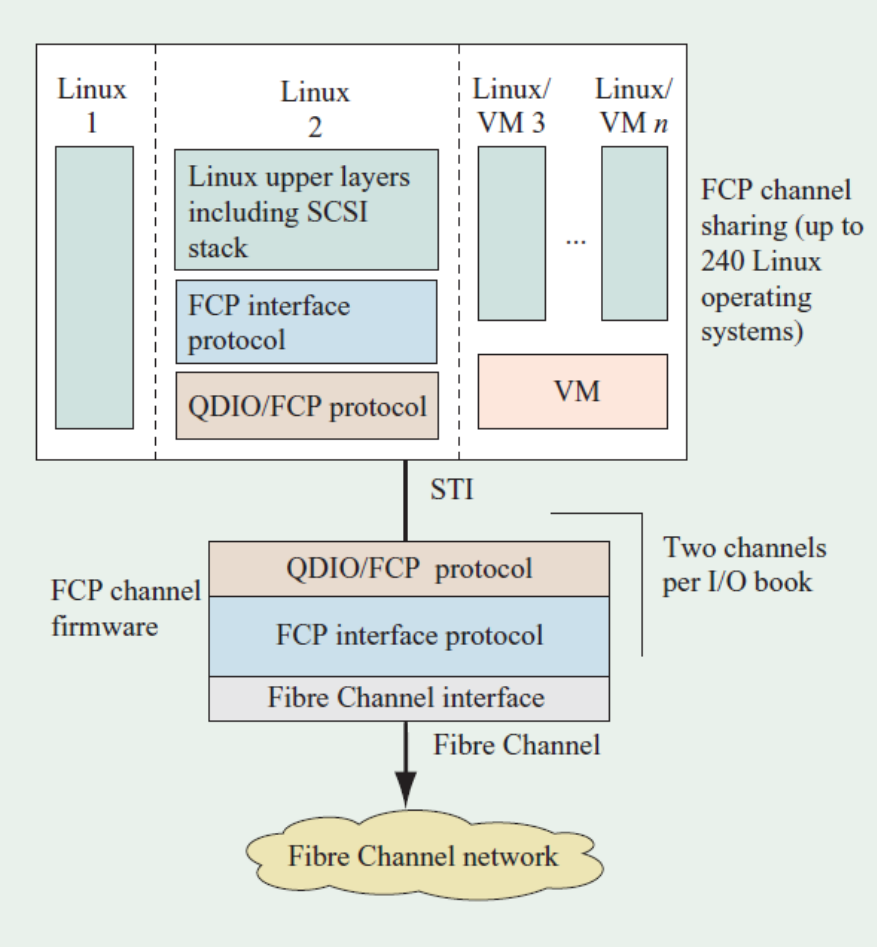


Device number definitions identify the communication path to the FCP channel and QDIO device queue.

WWPN (8 byte)-
World Wide Port Name is used to query the controller port connection in the fabric.

LUN (8 Byte)
Logical Unit Number
Identifies the device

FCP channel overview



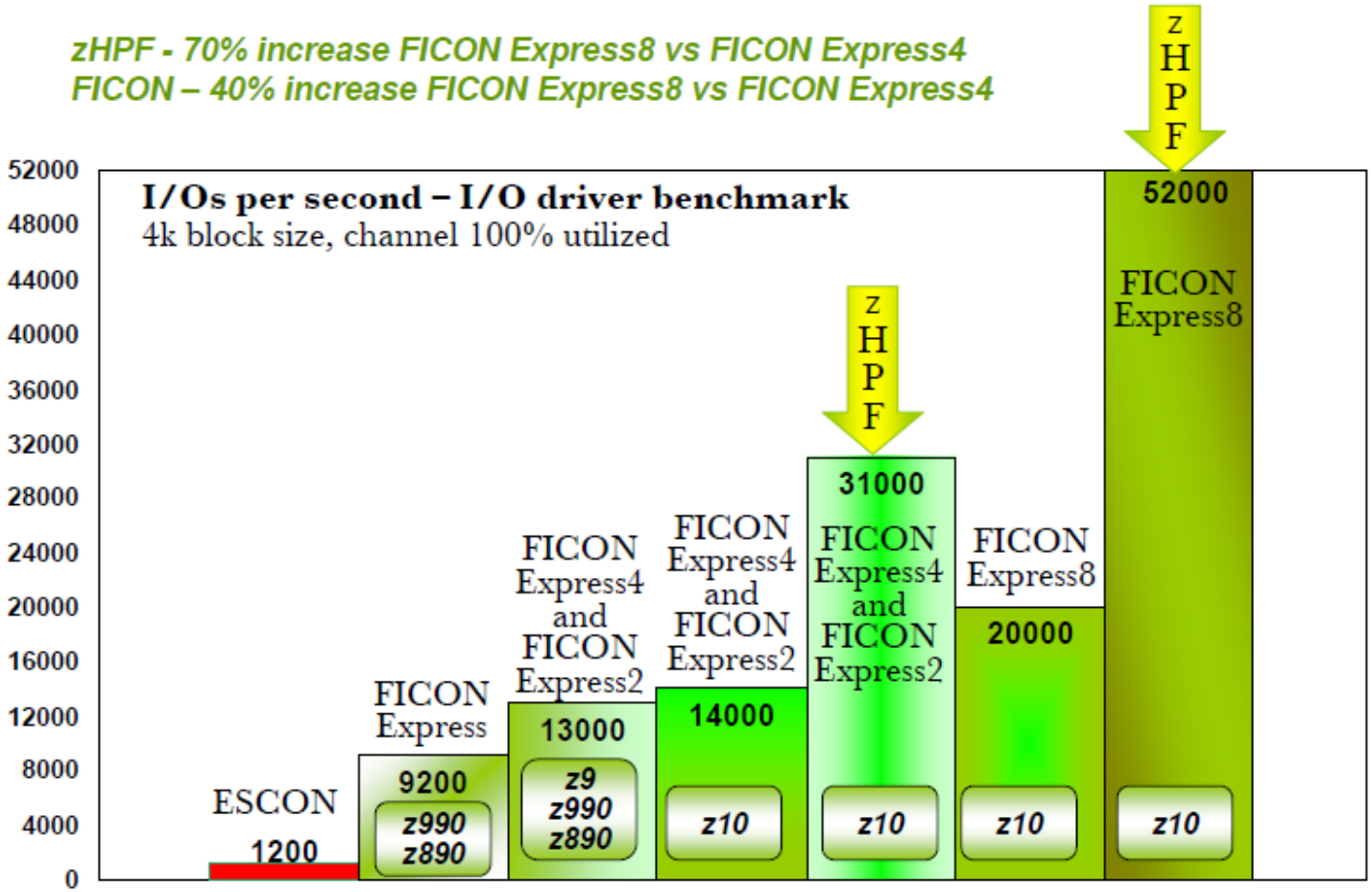
Reference: J. Srikrishnan, S. Amann, et al. “Sharing FCP Adapters Through Virtualization.” *IBM Journal of Research and Development.* 51 No. ½, 103-117 (2007).

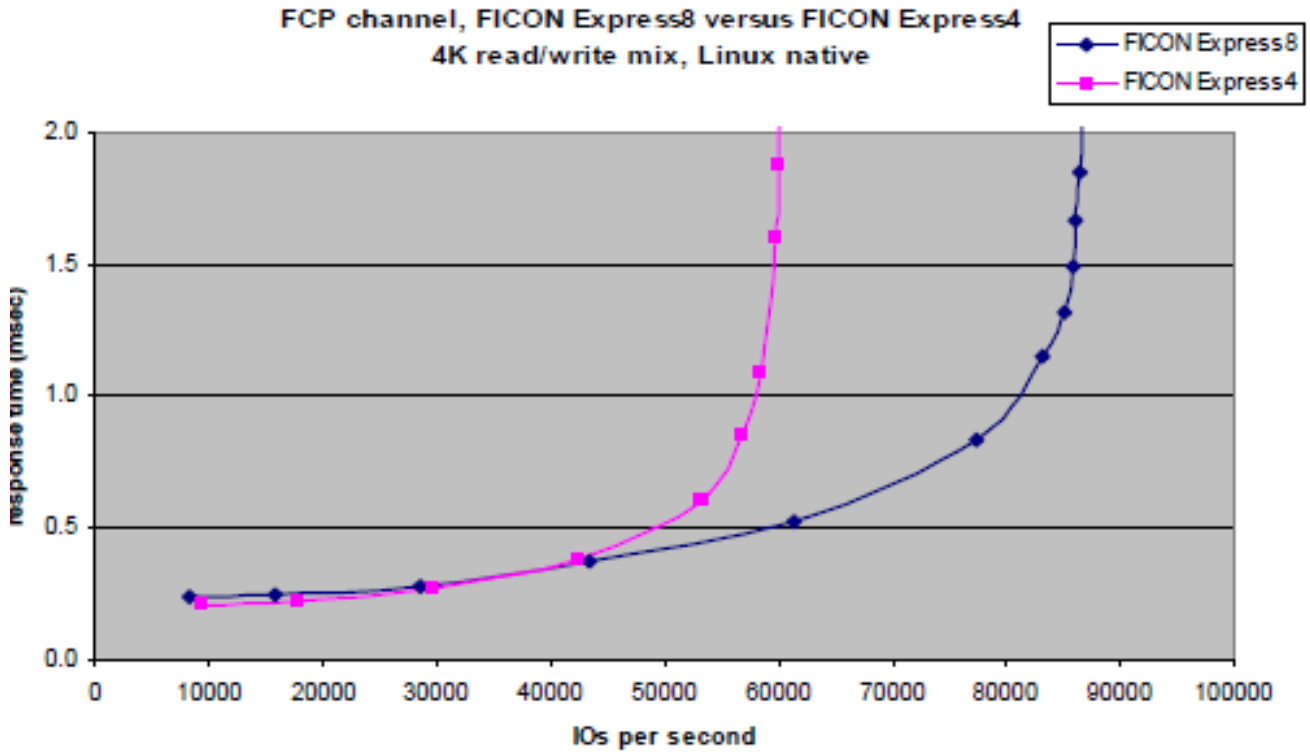
Graphics in this section from IBM performance documentation

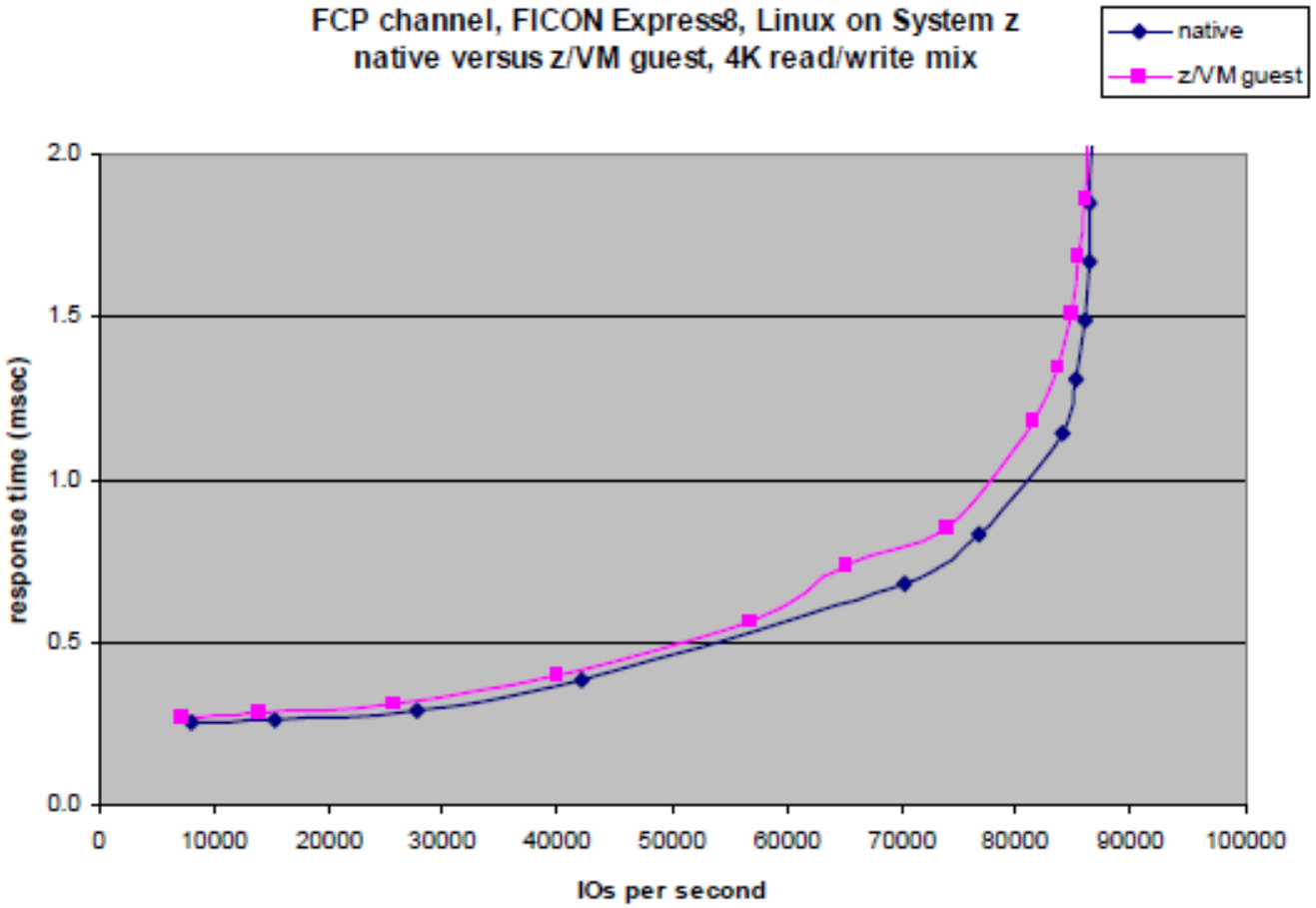
FCP CHANNEL PERFORMANCE

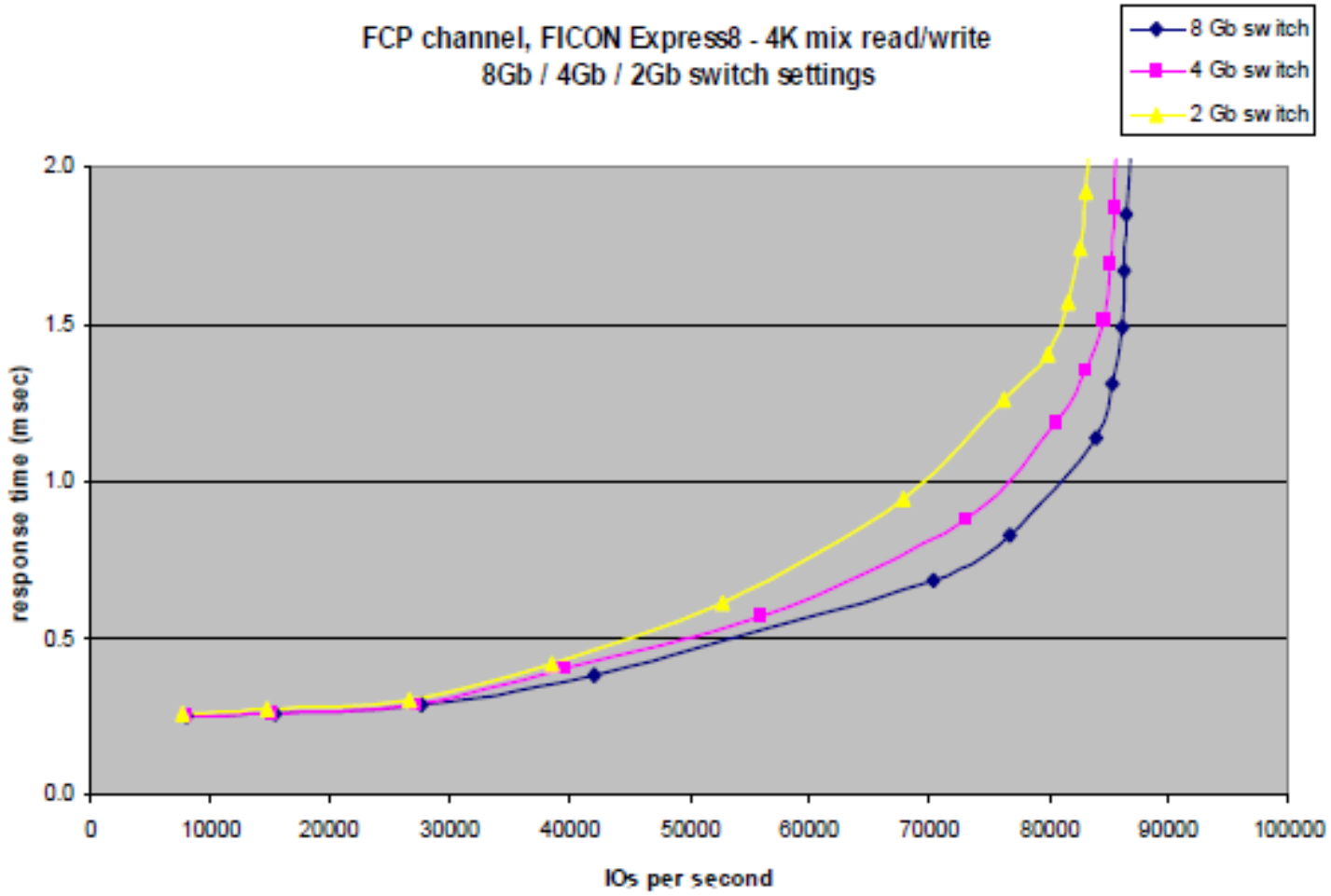
FICON Express 8 Performance Start I/Os

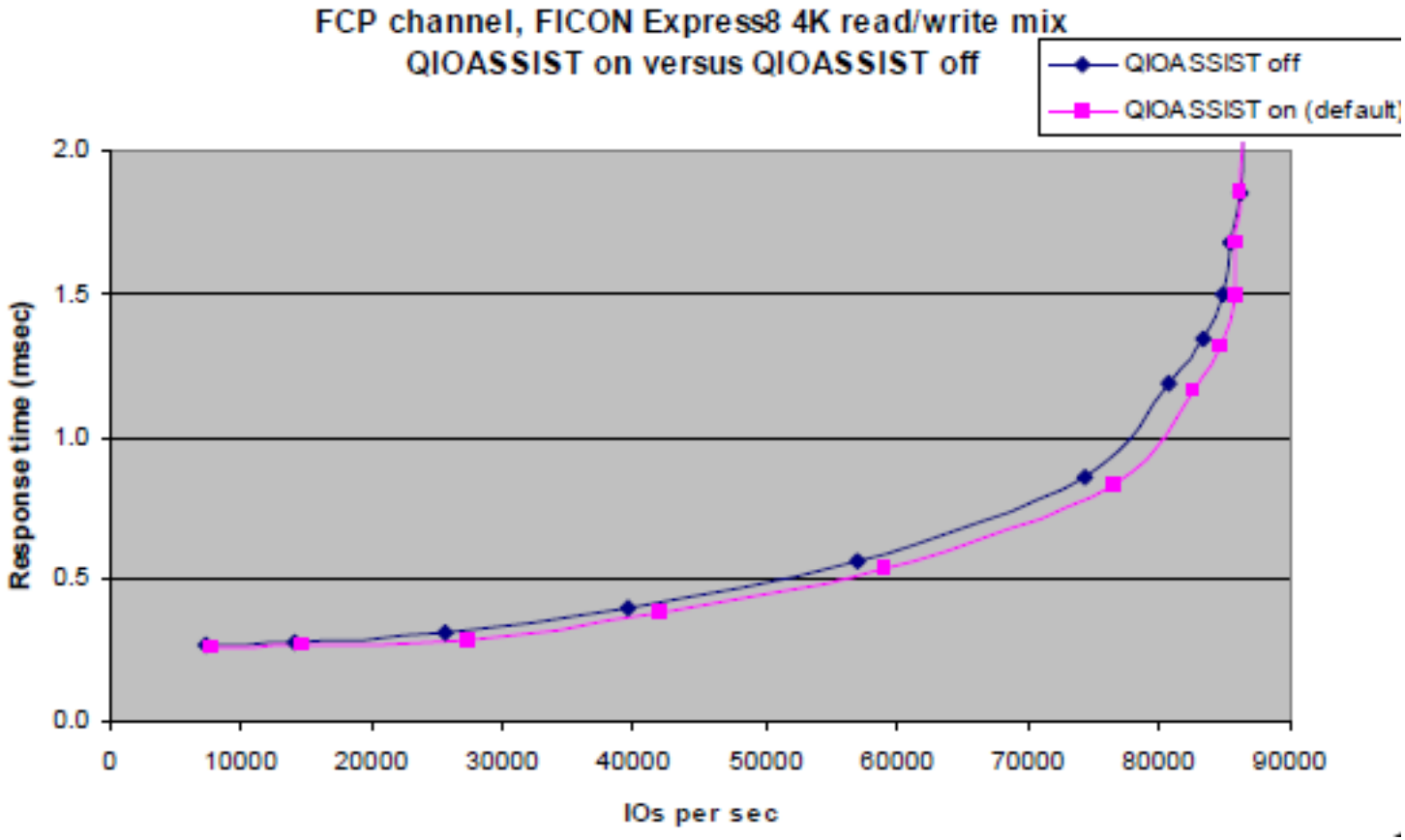
zHPF - 70% increase FICON Express8 vs FICON Express4
FICON – 40% increase FICON Express8 vs FICON Express4

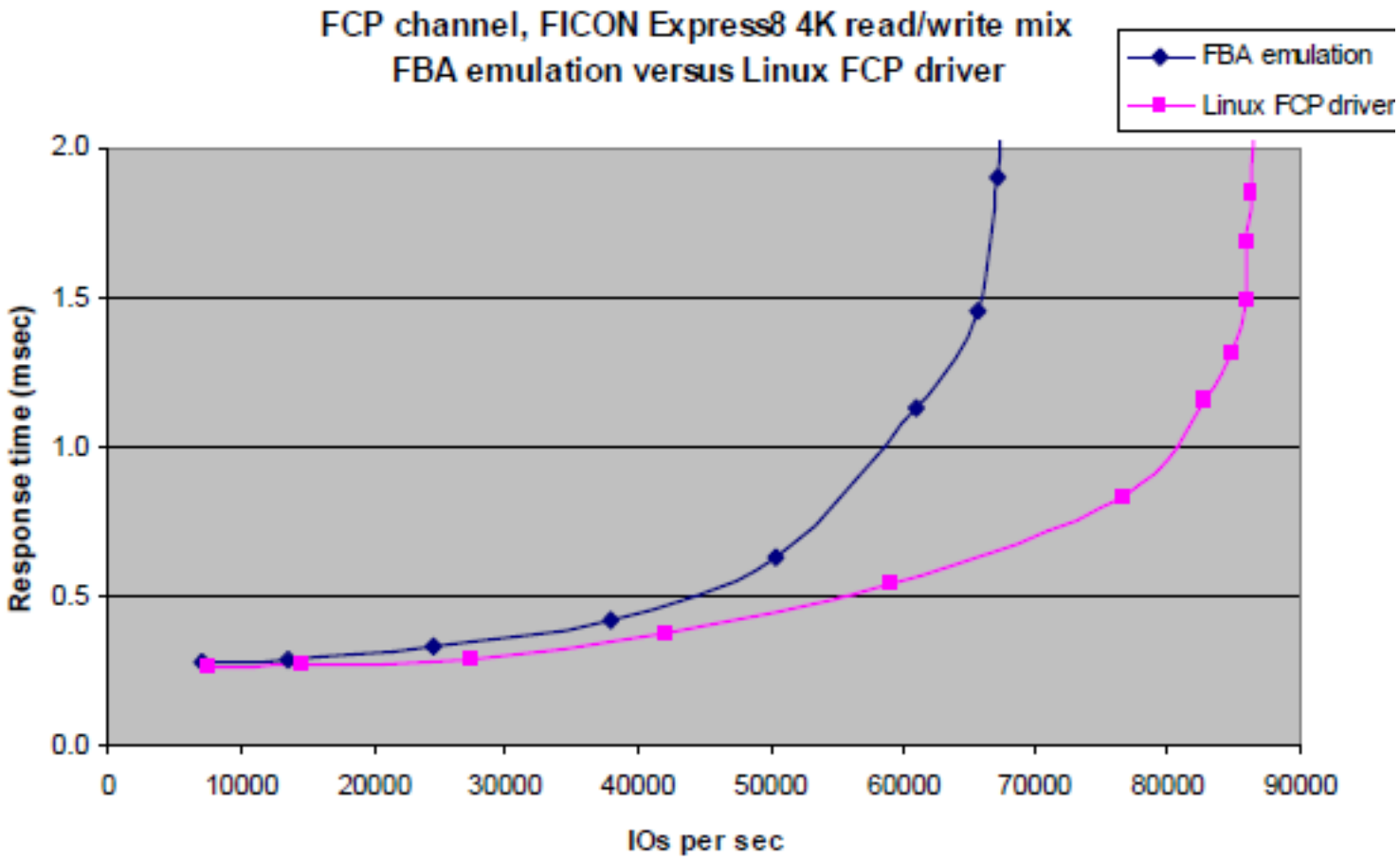



FCP channel, FICON Express8 versus FICON Express4
4K read/write mix, Linux native
FICON Express8
FICON Express4
response time (msec)
2.0
1.5
1.0
0.5
0.0
0
10000
20000
30000
40000
50000
60000
70000
80000
90000
100000
IOs per second


FCP channel, FICON Express8, Linux on System z
native versus z/VM guest, 4K read/write mix
native
z/VM guest
response time (msec)
2.0
1.5
1.0
0.5
0.0
0
10000
20000
30000
40000
50000
60000
70000
80000
90000
100000
IOs per second


FCP channel, FICON Express8 - 4K mix read/write
8Gb / 4Gb / 2Gb switch settings
8 Gb switch
4 Gb switch
2 Gb switch
response time (msec)
2.0
1.5
1.0
0.5
0.0
0
10000
20000
30000
40000
50000
60000
70000
80000
90000
100000
IOs per second


FCP channel, FICON Express8 4K read/write mix
QIOASSIST on versus QIOASSIST off
QIOASSIST off
QIOASSIST on (default)
Response time (msec)
2.0
1.5
1.0
0.5
0.0
0
10000
20000
30000
40000
50000
60000
70000
80000
90000
IOs per sec


FCP channel, FICON Express8 4K read/write mix
FBA emulation versus Linux FCP driver
FBA emulation
Linux FCP driver
Response time (msec)
2.0
1.5
1.0
0.5
0.0
0
10000
20000
30000
40000
50000
60000
70000
80000
90000
IOs per sec

Channel Path Activity

CHANNEL PATH ACTIVITY

z/OS V1R10 SYSTEM ID S01 DATE 07/27/2009 INTERVAL 23.58.731
RPT VERSION V1R10 RMF TIME 20.06.01 CYCLE 1.000 SECONDS

IODF = 01 CR-DATE: 08/06/2008 CR-TIME: 10.50.45 ACT: POR MODE: LPAR CPMF: EXTENDED MODE

DETAILS FOR ALL CHANNELS

CHANNEL PATH			UTILIZATION(%)			READ(MB/SEC)		WRITE(MB/SEC)		FICON OPERATIONS			ZHPF OPERATIONS		
ID	TYPE	G	SHR	PART	TOTAL	BUS	PART	TOTAL	PART	TOTAL	RATE	ACTIVE	DEFER	RATE	ACTIVE	DEFER
C0	FCP	9	Y	0.00	53.89	14.02	0.00	163.99	0.00	164.11						
C1	FCP	9	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00						
C2	FCP	9	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00						

- All FICON Express8 channels provide Channel Path Activity information, which includes FICON processor and bus utilization, and MBytes read and written. If z/OS is running in a separate LPAR from Linux on System z LPAR, z/OS can report FCP channel usage in the Linux LPAR

Performance notes

- FICON Express8 channels operating as FCP channels yield significant performance benefits
- Little channel I/O performance difference between Linux native vs VM guest
- Little difference whether QIO assist is on or off
- Available performance metrics tools:
  - IOSTAT
  - zFCP Device driver
- Additional detailed info:
  - "*Running Linux on IBM System z9® and IBM eServer™ zSeries® under z/VM" (SG24-6311).*

Part 2

CHANNEL SHARING AND MANAGEMENT CHALLENGES

FCP channel and device sharing

- An FCP channel can be shared between multiple Linux operating systems, each running in a logical partition or as a guest operating system under z/VM.
- To access the FCP channel, each operating system needs its own QDIO queue pair defined as a data device on an FCP channel in the HCD/IOCP.
- These devices are internal software constructs and have no relation to physical devices outside of the adapter.
- These QDIO devices are also referred to as subchannels.

FCP channel and device sharing

- An FCP channel can be shared between multiple Linux operating systems, each running in a logical partition or as a guest operating system under z/VM.
- To access the FCP channel, each operating system needs its own QDIO queue pair defined as a data device on an FCP channel in the HCD/IOCP.
- These devices are internal software constructs and have no relation to physical devices outside of the adapter.
- These QDIO devices are also referred to as subchannels.
- The host operating system uses these subchannels as vehicles to establish conduits to the FCP environment.
- Each subchannel represents a virtual FCP adapter that can be assigned to an operating system running either natively in an LPAR or as a guest OS under z/VM.

FCP channel and device sharing

- Initially, support was for up to 240 z/Architecture defined subchannels.
- Currently each FCP channel can support up to 480 subchannels/QDIO queue pairs.
  - Each FCP channel can be shared among 480 operating system instances, with the caveat of a maximum of 252 guests per LPAR).
- Host operating systems sharing access to an FCP channel can establish a total of up to 2048 concurrent connections to up to 512 different remote fibre channel ports associated with fibre channel controllers.
- Total number of concurrent connections to end devices, identified by logical unit numbers (LUNs) must not exceed 4096.
- WAT keeps track of it all

FCP channel sharing

- When NPIV is not implemented, multiple Linux images share an FCP channel and all of the Linux images have connectivity to all of the devices connected to the FCP fabric.
- Since the Linux images all share the same FCP channel, they all use the same WWPN to enter the fabric.
  - Makes them indistinguishable from each other within the fabric.
  - Hence the use of zoning in switches and LUN-masking in controllers cannot be effective.
  - Creates a management challenge

Management challenge

- When sharing an FCP adapter, System z must ensure the OS images sharing System z resources have the same level of protection and isolation as if each OS was running on its own dedicated server.
- For accessing storage devices via a shared host adapter, this means that the same level of access protection must be achieved as if each OS was using its own dedicated IO adapter.
- FCP LUN Access Control (pre System z9)
- NPIV

Integrating System z using z/OS, zLinux and Node Port ID Virtualization (NPIV)

zSeries/System z server virtualization

- zSeries/System z support of zLinux
  - Mainframe expanded to address open system applications
  - Linux promoted as alternative to Unix
  - Mainframe operating system virtualization benefits
    - Availability, serviceability, scalability, flexibility
- Initial zSeries limits
  - FCP requests are serialized by the operating system
    - FCP header does not provide image address
    - FICON SB2 header provides additional addressing
  - Channel ports are underutilized
  - Resulting cost/performance benefit is not competitive

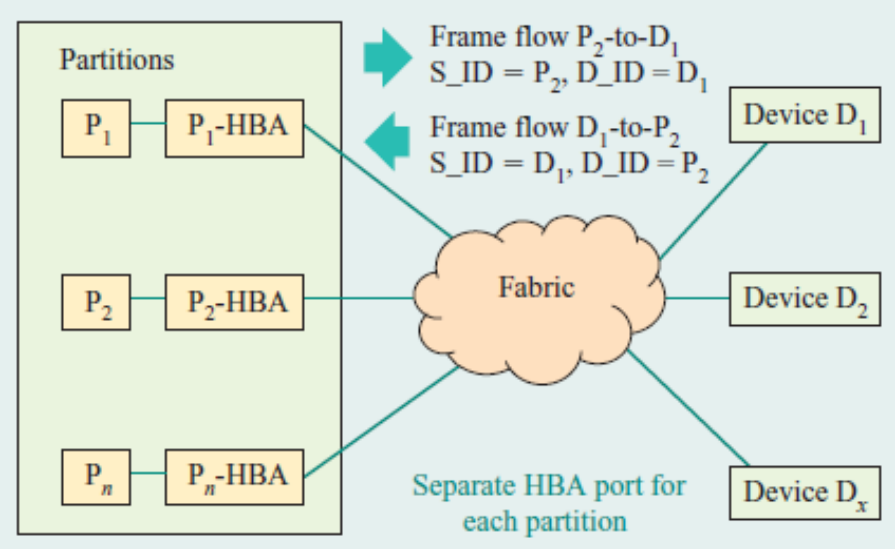
FCP Device sharing

- Without using NPIV, an FCP channel prevents logical units from being opened by more than one Linux image at a time.
  - Access is on a first come, first served basis.
  - This prevents problems with concurrent access from Linux images that are sharing the same FCP channel (same WWPN).
  - This usage serialization means that one Linux image can block other Linux images from accessing the data on one or more logical units, unless the sharing images (z/VM guests) are “well behaved” and not in contention.

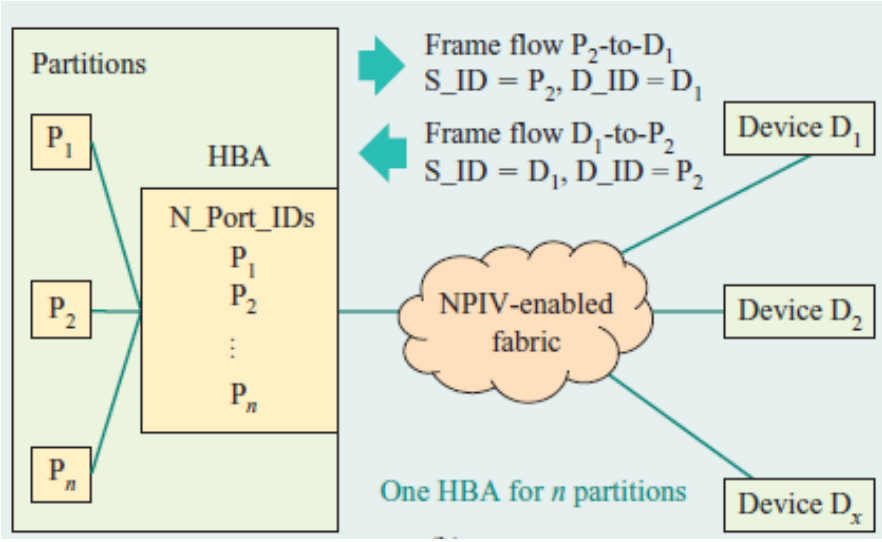
FCP channel/device sharing-summary

- Different host operating systems sharing access to a single FCP channel may access the same fibre channel port via this channel.
- While multiple operating systems can concurrently access the same remote fibre channel port via a single FCP channel, fibre channel devices (identified by their LUNs) can only be serially re-used.
- In order for two or more unique operating system instances to share concurrent access to a single fibre channel or SCSI device (LUN), each of these operating systems must access this device through a different FCP channel.
- Should two or more unique operating system instances attempt to share concurrent access to a single fibre channel or SCSI device (LUN) over the same FCP channel, a LUN sharing conflict will occur, resulting in errors.

With and without NPIV



Without NPIV



With NPIV

Reference: J. Srikrishnan, S. Amann, et al. “Sharing FCP Adapters Through Virtualization.” *IBM Journal of Research and Development.* 51 No. ½, 103-117 (2007).

NPIV

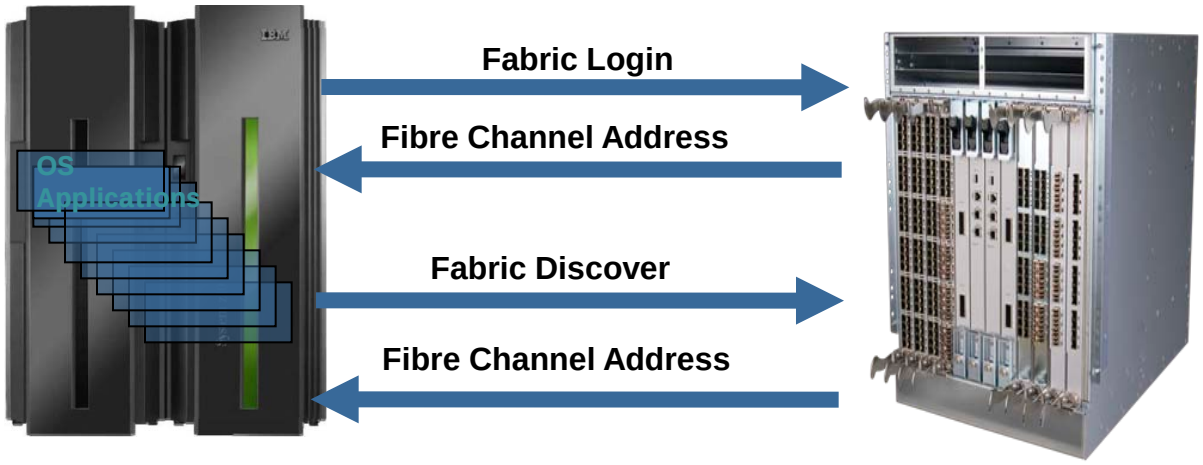
- NPIV is designed to allow the sharing of a single physical FCP channel among operating system images, whether it be in logical partitions (LPARs) or as z/VM guests in virtual machines.
- This is accomplished by assigning a unique World Wide Port Name (WWPN) to each operating system connected to the FCP channel.
- Each OS then appears to have its own distinct WWPN in a SAN environment, enabling traffic separation via zoning and LUN masking.

The road to NPIV-

- Alternatives looked at by IBM included:
  - FC process associators
  - Hunt groups/multicasting
  - Identifying the OS image at the upper level protocol layer
  - Emulating sub fabrics
- Finally settled on NPIV

Server Consolidation-NPIV

- N_Port Identifier Virtualization (NPIV)
  - Mainframe world: unique to System z9 and System z10
  - zLinux on System z9/10 in an LPAR
  - Guest of z/VM v 4.4, 5.1 and later
  - N_Port becomes virtualized
    - Supports multiple images behind a single N_Port
  - N_Port requests more than one FCID
    - FLOGI provides first address
    - FDISC provides additional addresses
  - All FCID's associated with one physical port



Node Port ID Virtualization (NPIV)

- Allows each operating system sharing an FCP channel to be assigned a unique virtual world wide port name (WWPN).
  - Used for both device level access control in a storage controller (LUN masking) and for switch level access control on a fibre channel director/switch (zoning).
- A single, physical FCP channel can be assigned to multiple WWPNs and appear as multiple channels to the external storage network.
- The virtualized FC Node Port IDs allow a physical fibre channel port to appear as multiple, distinct ports.
  - IO transactions are separately identified, managed, transmitted, and processed just as if each OS image had its own unique physical N port.

Using ELS to assign N_Port IDs

- FC standard defines a set of services used to establish communications parameters, each of which is called an *extended link service* (ELS).
- An ELS consists of a request sent by an N_Port and a response returned by a recipient port.
- One ELS, called a *fabric login* (FLOGI), is sent from an N_Port to its attached fabric port (F_Port) in the adjacent switch to request the assignment of an N_Port ID.

Using ELS to assign N_Port IDs: FLOGI

- The FLOGI request is the first frame sent from an N_Port to its adjacent switch.
- The purpose of the FLOGI ELS is to enable the switch and the N_Port to exchange initialization parameters
  - Includes unique identifiers known as *worldwide port names* (WWPNs)
  - Allows the fabric to assign an N_Port ID to the N_Port.
- The switch responds with the N_Port assigned to the requesting N_Port.
- Because the N_Port that sends the FLOGI request does not yet have an N_Port ID, it sets the S_ID in the FLOGI request to zero.
  - The switch responds with a FLOGI-accept response that contains the assigned N_Port ID.
  - The “HBA” uses this assigned N_Port ID as the S_ID when sending subsequent frames.

FLOGI (Cont'd)

- The N_Port ID assigned to a given N_Port may change each time the N_Port is reinitialized and performs the FLOGI ELS, but the WWPN of the N_Port does not change.
- This allows the fabric to more effectively manage N_Port ID assignments.
- Provides for persistent and repeatable recognition of the identity of an N_Port (WWPN) regardless of the physical fabric port it is attached to.
- N_Ports become associated with a specific OS image
  - The WWPN can be used to identify the owning OS and the access privileges it requires.

Need to request multiple N_Port IDs: FDISC

- Fabric Discovery (FDISC) is another ELS
- Original purpose is to verify an existing login with the fabric is still valid.
- The FDISC was always sent with a non-zero S_ID (the presumed S_ID of the sender).
- This made it possible to obtain additional N_Port IDs by an extension of the FDISC ELS.
- An unlimited number of additional N_Port IDs could be obtained by using the following protocol:

Need to request multiple N_Port IDs: FDISC(2)

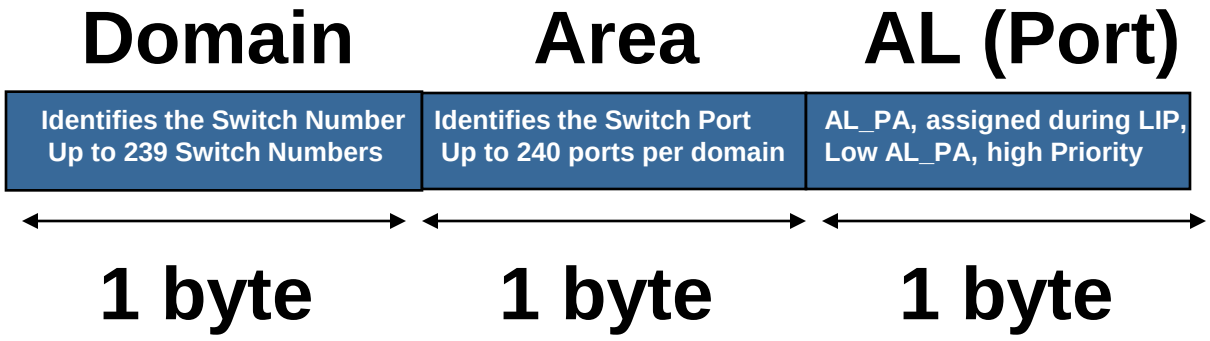
- The FLOGI request is sent (with the S_ID set to zero and the WWPN set to that of the first OS image requiring an N_Port ID) to request the first N_Port ID.
- The fabric assigns the first N_Port ID to the N_Port, and that N_Port ID is used by the first OS image.
- The FDISC request is sent (with the S_ID set to zero and the WWPN set to that of the next OS image requiring an N_Port ID) to request the next N_Port ID.
- The fabric assignes the next N_Port ID to the N_Port, and that N_Port ID is used by the next OS image.

FDISC (3)

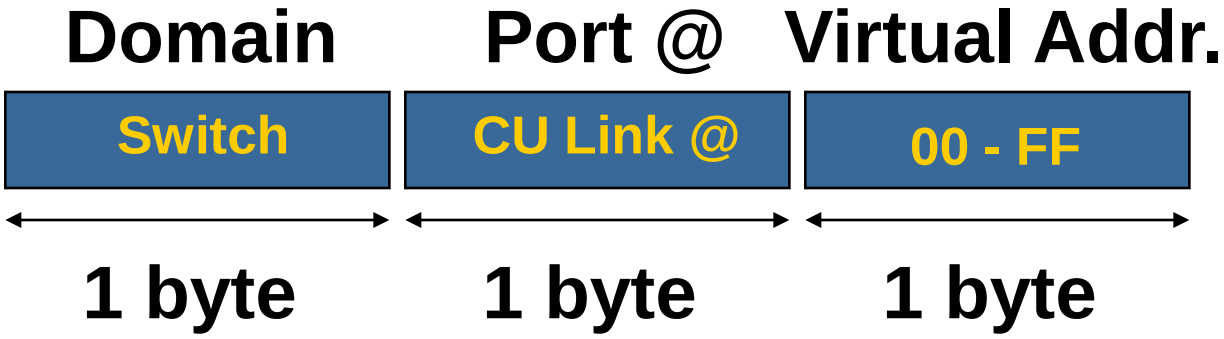
- The FDISC request can be repeated until all N_Port IDs have been assigned to the N_Port.
- It may also be executed in parallel in order to decrease the time taken to obtain all N_Port IDs.
- This protocol for obtaining the additional N_Port IDs has been incorporated into the FC standards and is referred to as NPIV.

System z N-port ID Virtualization

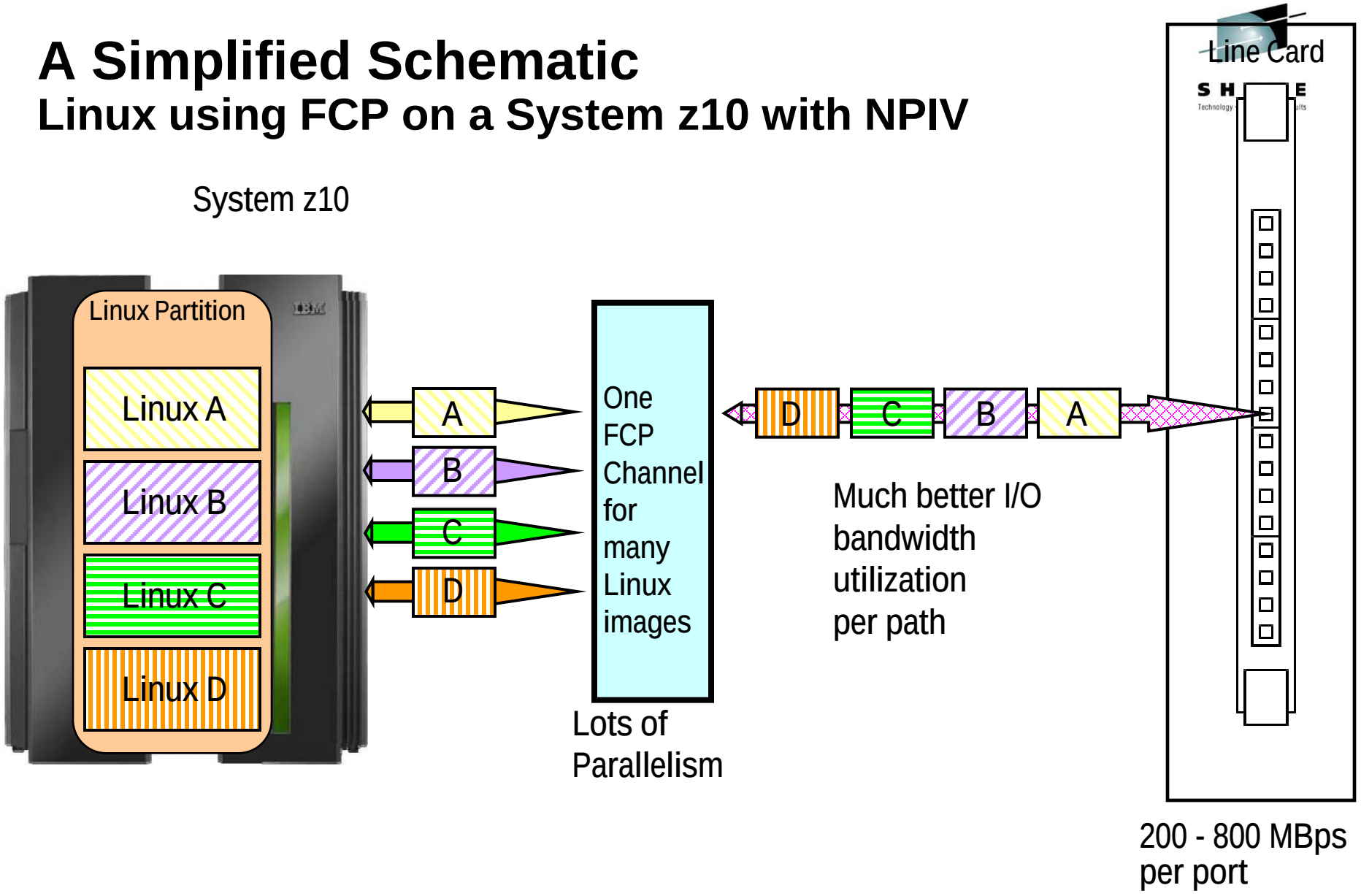
FC-FS 24 bit fabric addressing – Destination ID (D_ID)



FICON Express2, Express4 and Express 8 adapters now support NPIV



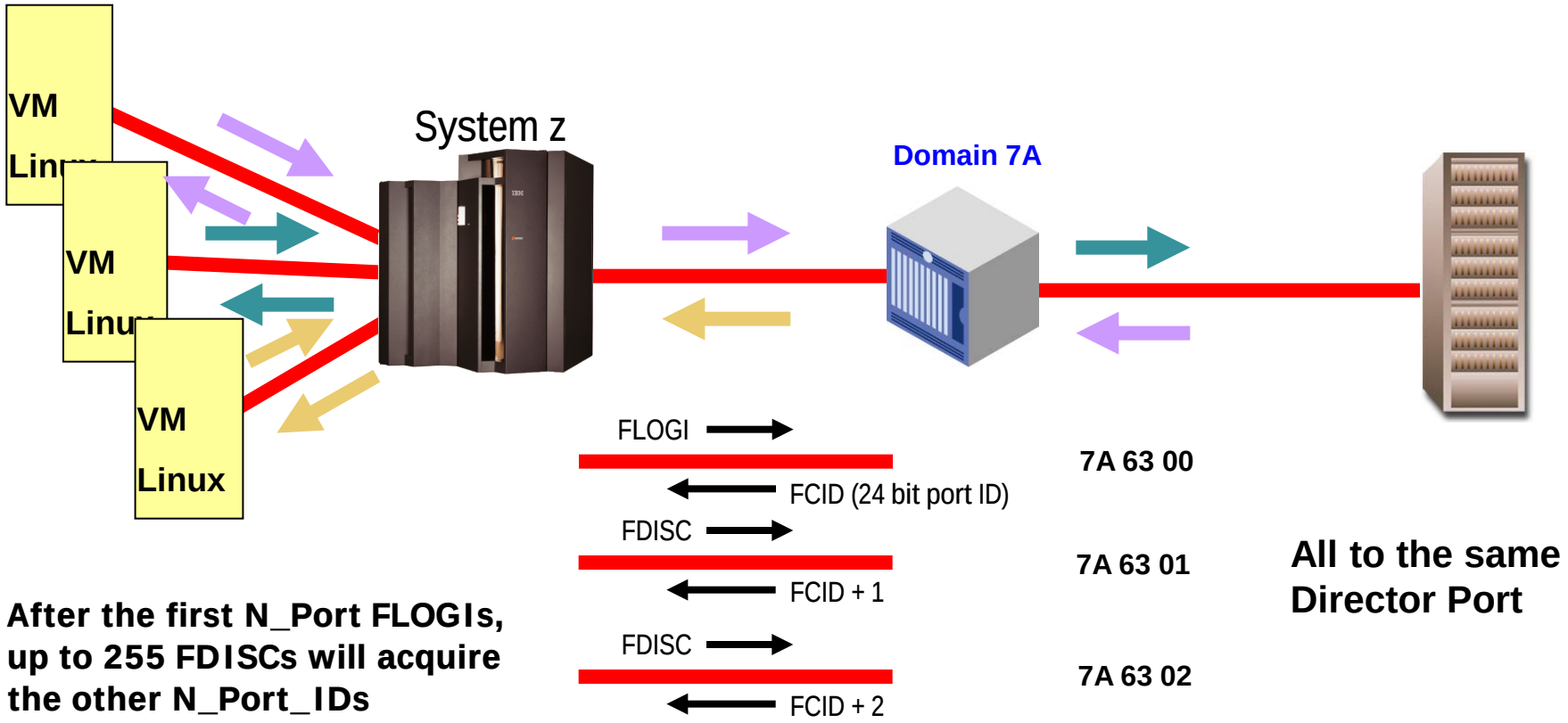
A Simplified Schematic
Linux using FCP on a System z10 with NPIV



System z N-port ID Virtualization-summary

One System z server port can have up to 255 NP-IDs
•IBM has told us it wants this expandable to thousands

- NPIV on the System z
  - FCP Driver for System z
  - Same CHPIDs as used for FICON



Etc. for up to 256 FCIDs per channel path

NPIV summary

- NPIV allows multiple zLinux “servers” to share a single fibre channel port
  - Maximizes asset utilization
    - Open systems server ROT is 10 MB/second
    - 4 Gbps link should support 40 zLinux “servers” from a bandwidth perspective
- NPIV is an industry standard

Standards and NPIV

- FC-FS
  - Describes FDISC use to allocate additional N_Port_IDs
    - Section 12.3.2.41
  - NV_Ports are treated like any other port
    - Exception is they use FDISC instead of FLOGI
- FC-GS-4
  - Describes
    - Permanent Port Name and Get Permanent Port Name command
      - *Based on the N_Port ID (G_PPN_ID)*
    - The PPN may be the F_Port Name
- FC-LS
  - Documents the responses to NV_Port related ELSs
    - FDISC, FLOGI and FLOGO
    - Reference 03-338v1

More Standards on NPIV

- FC-DA
  - Profiles the process of acquiring additional N_Port_IDs
    - Clause 4.9
- FC-MI-2
  - Profiles how the fabric handles NPIV requests
    - New Service Parameters are defined in 03-323v1
    - Name Server Objects in 7.3.2.2 and 7.3.2.3