



**z/OS® V1R10 Communications Server**

***Hardware exploitation***

@business on demand software

© 2008 IBM Corporation

This presentation covers changes made to Communications Server to exploit z hardware improvements.

## IPv4 INTERFACE statement for QDIO

- New for IPv4 OSA-Express QDIO (IPAQENET)
- Patterned after IPv6 INTERFACE statement
- Enhances usability
  - ▶ Combines function of DEVICE/LINK/HOME
  - ▶ Easier to add/modify/delete
  - ▶ Improved source VIPA specification
  - ▶ Provides control over VIPA ARP processing
- DEVICE/LINK still supported

New for is an IPv4 INTERFACE statement for IPAQENET. This is the only interface type supported by the IPv4 Interface statement. This statement is similar to the IPv6 INTERFACE statement for IPAQENET6 in that it provides the equivalent of DEVICE/LINK/HOME definitions in one statement. For example, it uses the PORTNAME parameter to identify the TRLE and uses the SOURCEVIPAINTERFACE parameter for the source VIPA function.

There are some differences between IPv4 and IPv6 Interface statements. The IPv4 statement does not contain any of the IPv6-specific parameters (such as ADDADDR, DELADDR, DEPRADDR, DUPADDRDET, INTFID). The IPv4 statement adds the IPv4-specific IPBCAST parameter from the IPAQENET LINK statement. For IPv4, IPADDR is required and must be a single IP address with an optional subnet mask definition. For IPv6, IPADDR is optional and can contain multiple IP addresses.

Without this support, adding or deleting IPv4 QDIO interfaces using OBEYFILE can be quite cumbersome because the HOME statement is a complete replacement for the existing home list. This task is much easier with the INTERFACE statement.

The INTERFACE statement also provides an improved method for specifying the source virtual IP address (VIPA) and provides control over VIPA Address Resolution Protocol (ARP) processing.

You can still use DEVICE/LINK for IPv4 QDIO. You can also intermix DEVICE/LINK and IPv4 INTERFACE in the same configuration for different IPv4 interfaces.

## IPv4 INTERFACE statement example

```

INTERFACE NSQDIO411 DEFINE IPAQENET
  IPADDR 172.16.11.1 /24
  PORTNAME NSQDIO1
  MTU 1492
  VMAC
  SOURCEVIPAINTERFACE LVIPAI

```

Subnet mask  
255.255.255.0  
specified (controls  
VIPA ARPs)

OSA:  
Generate  
a virtual  
MAC  
address

Link name of the static  
VIPA to be used for the  
SOURCEVIPA function  
on this interface

This slide shows an example of an IPv4 interface statement IPAQENET. This statement defines an IPv4 interface associated with the OSA-Express port NSQDIO1. It specifies a subnet mask of 24 bits ('FFFFFF00'x) and defines a unique subnet. The statement requests that Open Systems Adapter (OSA) generate a virtual Media Access Control (MAC) address (and defaults to ROUTEALL) and specifies the link\_name of a static VIPA for the source VIPA function.

Specifying the subnet mask on the Ipv4 interface statement reduces Address Resolution Protocol (ARP) traffic on your network. It prevents the TCP/IP stack from performing gratuitous ARPs or responding to ARP requests for VIPAs unless the VIPA is in the same subnet as the interface. Previously TCP/IP did gratuitous ARP or responded to ARP requests for all VIPAs, which can get burdensome if there are large numbers of VIPAs or DVIPAs on this stack.

## Effect of IPv4 Interface statement on other statements and commands

- Specify IPv4 interface name in place of device or link name on:
  - ▶ START/STOP (including VARY command)
  - ▶ ROUTE (in BEGINROUTES block)
  - ▶ PRIMARYINTERFACE
  - ▶ PKTTRACE (including VARY command)
  - ▶ VARY TCPIP,,PURGECACHE
  - ▶ Netstat ARp/-R
- Others restricted to LINK definitions:
  - ▶ HOME, GATEWAY
  - ▶ BSDROUTINGPARMS

You can specify an IPv4 interface name on several statements and commands where you specify a device name (for IPv4 DEVICE/LINK/HOME) or an IPv6 interface name. They are START (including VARY TCPIP,START command) and STOP (including VARY TCPIP,STOP command)

On several statements and commands, you can specify an IPv4 interface name where you can currently specify an IPv4 link name. You can do this on ROUTE (in BEGINROUTES block), PRIMARYINTERFACE, PKTTRACE (including VARY TCPIP,,PACKETTRACE command), VARY TCPIP,,PURGECACHE , and Netstat ARp/-R

On several statements, you cannot to specify an IPv4 interface name. These statements are restricted to IPv4 interfaces defined with the LINK statement. The HOME statement is only allowed with DEVICE and LINK, because you specify home address on INTERFACE statement. The GATEWAY statement is only allowed with DEVICE and LINK, you should migrate to BEGINROUTES instead. Finally, BSDROUTINGPARMS is not compatible with the INTERFACE statement because you can configure MTU and subnet mask directly on the INTERFACE statement.

## Converting to IPv4 Interface statement

- Each INTERFACE definition requires a separate DATAPATH device
- OBEYFILE restriction
  - ▶ When OSA also configured for IPv6, need to change interface name (or restart stack with new profile)
- OMPROUTE's existing MTU and subnet mask parameters
  - ▶ OMPROUTE checks for mismatches with INTERFACE statement in stack profile
    - Issues new message (EZZ8163I or EZZ8164I) and uses value configured to OMPROUTE
- Be aware of netstat, SNMP, message changes

If you convert from DEVICE/LINK to INTERFACE for IPv4 and you also have an IPv6 definition for the same OSA, the stack uses a separate DATAPATH device for each interface. You need to ensure that the TRLE definition has enough DATAPATH devices.

If the OSA is also configured for IPv6, you cannot convert from a DEVICE/LINK definition to an IPv4 INTERFACE definition in an OBEYFILE and re-use the link name as the interface name. In this scenario, you either need to use a different name for the INTERFACE or restart the stack with the new profile. This restriction does not apply if you did not also configure an IPv6 definition for that OSA.

The IPv4 INTERFACE statement contains subnet mask and MTU parameters which are also configurable on these existing OMPROUTE configuration statements: OSPF\_INTERFACE, RIP\_INTERFACE, INTERFACE.

OMPROUTE will detect a mismatch with the MTU or subnet mask values between the INTERFACE statement in the stack profile and the corresponding statement in the OMPROUTE configuration. If this mismatch is detected, OMPROUTE issues a new message and uses the value configured to OMPROUTE.

There are several Netstat changes.

The Netstat H0me/-h report splits IPv4 addresses into two sections, separating out those defined with the INTERFACE statement.

The Netstat DEvlinks/-d report now displays each IPv6 interface defined by an INTERFACE statement separately from the corresponding IPv4 interface defined by a DEVICE/LINK statement. The report also displays all interfaces which were defined with the INTERFACE statement using an IntfName heading as the first line. In V1R9 the IPv6 interfaces are displayed using a DevName heading as the first line (and were grouped underneath the DEVICE/LINK information of the corresponding IPv4 interface).

The Netstat DEvlinks/-d report now displays the configured VLAN ID for an inactive interface. Previously, this report only showed VLAN IDs for active interfaces.

The Netstat DEvlinks/-d report includes the OSA-Express portname and the DATAPATH device for OSA-Express QDIO interfaces which are defined with the INTERFACE statement. For IPv4 OSA-Express QDIO interfaces defined with DEVICE/LINK, the portname is already displayed as the device name, but the DATAPATH device is not displayed. For IPv6 HiperSockets™, the report now includes the CHPID. In V1R9, you need to look at the DevName field to identify the CHPID. For MPCPTP6, the report now includes the TRLE. In V1R9 you need to look at the DevName field to identify the TRLE.

Several messages that give IPv4 device or link names now have new equivalent messages that give the interface name

## VLAN background

- A Virtual LAN (VLAN) is a logical grouping of nodes using switches
- Nodes can communicate as if on same LAN
- Important factor in network planning
  - ▶ Can improve performance by reducing traffic on a physical LAN
  - ▶ Can enhance security by isolating traffic
  - ▶ Provides more flexibility in configuring networks

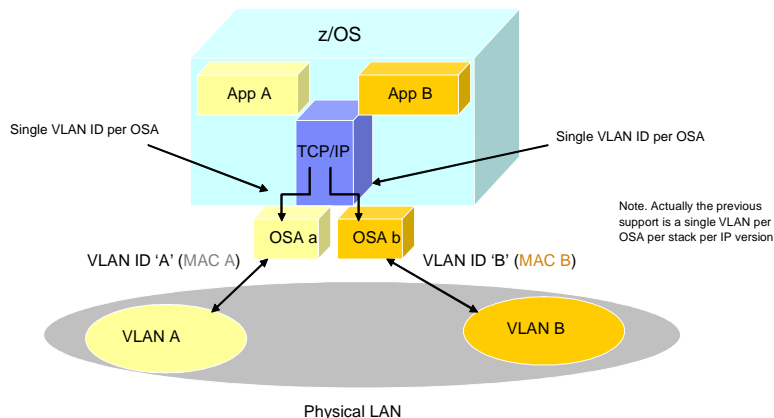
Virtual Local Area Network (VLAN) technology is becoming more important in network planning for many installations.

A LAN is a broadcast domain. Nodes on a LAN can communicate with each other without a router. You need a router to communicate across LANs.

A VLAN is a configured logical grouping of nodes using switches. Nodes on a VLAN can communicate as if they were on the same LAN. You need a router to communicate across VLANs

A VLAN configuration provides many benefits. VLANs can improve network performance by reducing traffic on a physical LAN. VLANs can enhance security by isolating traffic. VLANs provide more flexibility in configuring networks.

## V1R9 support: Single VLAN per OSA



7

Hardware exploitation

© 2008 IBM Corporation

This picture on this slide shows one physical LAN subdivided into two virtual LANs (VLANs). To configure VLAN for an OSA-Express in QDIO mode, you specify a VLAN ID in the TCP/IP profile. In V1R9 and earlier releases, the only way to access multiple VLANs from a given z/OS stack is to use multiple OSAs.

The z/OS stack registers the VLAN ID to OSA which causes OSA to do these steps. OSA appends a VLAN tag with this VLAN ID on all outbound packets (Note: For IPv6 unicast packets, the stack appends the VLAN tags). OSA also filters out any inbound packets which have a VLAN tag containing a different VLAN ID.

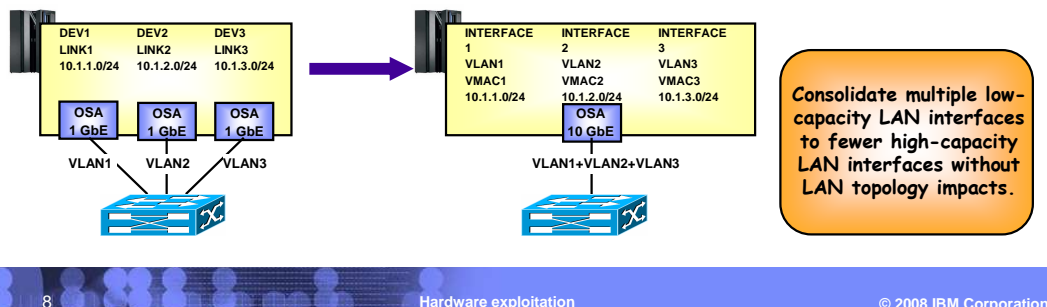
The V1R9 VLAN support in the z/OS stack is limited to one VLAN per OSA per IP version. IP version means IPv4 or IPv6. This inhibits a customer from consolidating OSA ports across different VLANs into a single port.

The only way to access multiple VLANs from a single z/OS stack is to use multiple OSAs.

The only way to access multiple VLANs through a single OSA is to use multiple stacks sharing the OSA.

## Multiple VLAN per OSA per stack per IP version

- V1R9 TCP/IP stack support is two VLANs per OSA port
  - One each for IPv4 and IPv6
- With consolidation to higher bandwidth OSA-Express (for example, 10 Gig) this is too restrictive
- V1R10: Support added for up to eight VLANs per IP protocol per OSA port for one TCP/IP stack



8

Hardware exploitation

© 2008 IBM Corporation

The V1R9 limitation of one VLAN per IP protocol is too restrictive because it is not possible to retain existing network interface and IP subnet topology when consolidating multiple LANs to one LAN. This requires IP renumbering.

In the picture on this slide, there are three VLANs each served by a 1 gigabit OSA. You want to consolidate these three 1 gigabit interfaces into 1 ten gigabit OSA. This would give you more bandwidth and a simpler configuration. However, without multiple VLAN support, you would have to renumber this network to merge the three 1-gigabit subnets into a single subnet for the 10 gigabit interface. With support for multiple VLANs over an OSA, you can now make this consolidation without renumbering the IP network. The single 10 gig OSA interface accesses the same three VLANs that the three 1 gigabit interfaces accessed.

With the new support, each VLAN on the same OSA port must use unique, non-overlapping IP subnets or prefixes. This is enforced by the TCP/IP stack.

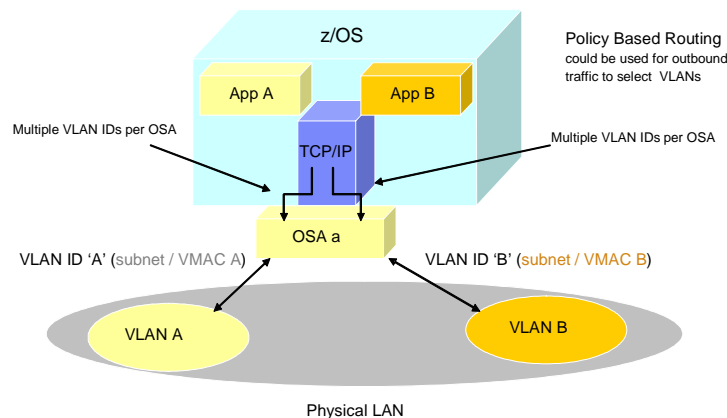
Each VLAN must be defined using a new IPv4-enabled version of the INTERFACE configuration statement, which only supports QDIO interfaces.

Each VLAN must use layer-3 virtual MAC addresses and each VLAN must have a unique MAC address.



## Multiple VLAN per OSA per stack per IP version

- Allows one stack to access multiple VLANs through the same OSA-Express feature
  - Multiple interfaces for same OSA-Express port
  - Maximum of eight IPv4 and eight IPv6
  - Requires use of virtual MACs (VMACs)



With V1R10, you can now configure multiple VLANs for the same OSA-Express feature (up to eight for IPv4 and eight for IPv6) from the same z/OS stack. You do this by defining multiple interfaces to the OSA-Express, one for each VLAN ID.

This solution provides several benefits.

**OSA port consolidation** - The multiple VLAN function allows you to consolidate multiple OSAs (for example, three 1Gb OSA ports) into a single OSA (for example, one 10Gb OSA port) serving multiple VLANs.

**Server consolidation** - The multiple VLAN function allows you to consolidate multiple application servers across multiple stacks into a single z/OS image where the traffic related to these servers is on unique VLANs.

**Improved quality of service with Policy-based routing** - The V1R9 policy-based routing function allows a z/OS stack to make routing decisions for IPv4 traffic that take into account additional criteria. These criteria can include job name, source port, destination port, protocol type (TCP or UDP), source IP address, NetAccess security zone, and multilevel secure environment security label. This enables you to route traffic meeting a certain criteria to one VLAN and traffic meeting a different criteria to another VLAN. However, in order to do this in V1R9 you need multiple OSAs (one per VLAN). The multiple VLAN function allows you to accomplish this with only one OSA

## Multiple VLAN configuration rules

- Must use INTERFACE statement to configure the interfaces
  - ▶ New for IPv4 (for OSA-Express QDIO)
  - ▶ No Multiple VLAN support for DEVICE/LINK
- Unique VLAN ID in stack profile
  - ▶ Within each IP version
- Use Virtual MAC (VMAC) with ROUTEALL
- Specify a unique subnet for each IPv4 definition

To define multiple interfaces to the same OSA-Express, you must follow all of these rules.

Configure each IPv4 interface for this OSA-Express feature in the TCP/IP profile using the INTERFACE statement for IPAQENET rather than DEVICE/LINK/HOME. Configure each IPv6 interface using the INTERFACE statement for IPAQENET6.

Configure a VLANID value on each IPv4 and each IPv6 INTERFACE statement for this OSA. Within each IP version, VLANID values must be unique.

Configure the VMAC parameter on each of these INTERFACE statements with the default ROUTEALL attribute. The VMAC address can either be specified or OSA-generated. If you specify a VMAC address, it must be unique for each INTERFACE statement.

Configure a unique subnet for each IPv4 interface for this OSA-Express feature using the subnet mask specification on the IPADDR parameter on the INTERFACE statement.

## Multiple VLAN IPv4 INTERFACE statement example

```

INTERFACE NSQDIO411 DEFINE IPAQENET
IPADDR 172.16.11.1/24
PORTNAME NSQDIO1
VLANID 411
MTU 1492
VMAC
SOURCEVIPAINTERFACE LVIPA1

INTERFACE NSQDIO412 DEFINE IPAQENET
IPADDR 172.16.1.1/24
PORTNAME NSQDIO1
VLANID 412
MTU 1492
VMAC
SOURCEVIPAINTERFACE LVIPA2
  
```

Different  
VLAN ids

Unique  
Subnets

Same  
OSA  
port

This is an example of multiple VLAN definitions with two INTERFACE statements for IPAQENET. Each statement defines an IPv4 interface associated with the same OSA-Express port NSQDIO1. Each specifies a subnet mask of 24 bits ('FFFFFF00'x) and defines a unique subnet. The statements contain different VLAN IDs, and each requests that OSA generate a virtual MAC address (and defaults to ROUTEALL). Each statement specifies the link\_name of a static VIPA for the source VIPA function.

## Multiple VLAN example: TRLE definition

```
TRLQDIO  TRLE  LNCTL=MPC ,          *
          READ=( 0E28 ) ,          *
          WRITE=( 0E29 ) ,         *
          MPCLEVEL=QDIO ,         *
          DATAPATH=( 0E2A-0E2F ) ,   *
          PORTNAME=NSQDIO1
```

Note 6 datapath devices.  
Every INTERFACE for this  
OSA will consume one.

This is an example TRLE definition for OSA-Express QDIO. While there are no changes to the TRLE definition, note that you need one DATAPATH device for each INTERFACE you configure for the OSA. This definition contains six DATAPATH devices.

## Multiple VLAN hardware requirements

- Ensure TRLE has enough DATAPATH devices
- Understand fixed storage requirements
  - ▶ Each DATAPATH device consumes fixed storage
- Ensure OSA-Express microcode level is current
  - ▶ Supports Layer 3 Virtual MAC address
  - ▶ Supports greater than 16 DATAPATH devices if needed
  - ▶ Refer to the 2094DEVICE and 2096DEVICE PSP buckets for details

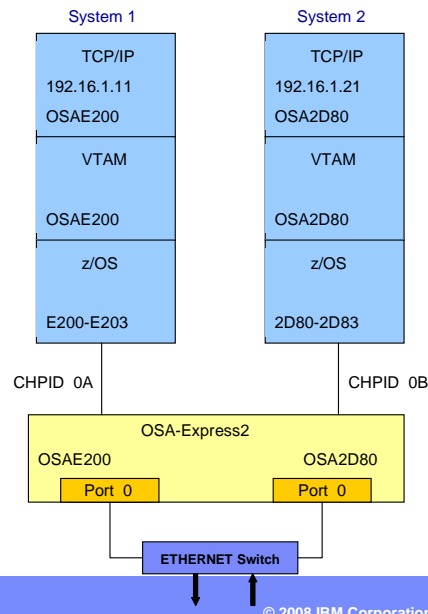
In order to use multiple VLANs for an OSA, you need to configure a separate interface to the OSA for each VLAN. Each of these interfaces requires a separate DATAPATH device in the TRLE definition. Furthermore, each DATAPATH device requires a certain amount of fixed storage. You can control the amount of fixed storage using the QDIOSTG VTAM<sup>®</sup> start option and the READSTORAGE parameter on the INTERFACE statement.

If you also configure the OSA-Express Network Traffic Analyzer (OSAENTA) function to trace OSA packets, this also uses a DATAPATH device. So, with eight IPv4 VLANs, eight IPv6, plus OSAENTA, a z/OS stack can have up to 17 DATAPATH devices for a single OSA. With eight stacks (and the restriction of only one OSAENTA per OSA), an LPAR can have up to 129 DATAPATH devices for a single OSA.

The multiple VLAN function is limited to OSA-Express Ethernet features in QDIO mode (CHPID type OSD) which support the Layer 3 Virtual MAC address (VMAC) function. Therefore, you must be running at a minimum with an IBM System z9 EC or z9 Business Class and an OSA-Express or OSA-Express2. You also need a certain OSA microcode level if you need to define more than 16 DATAPATH devices for a single OSA-Express port. Refer to the 2094DEVICE Preventive Service Planning (PSP) and the 2096DEVICE Preventive Service Planning (PSP) buckets for further information.

## OSA-Express3 multiport OSA support background information

- OSA-Express2 hardware
  - ▶ Single port per Channel Path ID (CHPID)



14

Hardware exploitation

© 2008 IBM Corporation

The OSA-Express2 OSA platform only supports a single port per Channel Path ID (CHPID). This OSA-Express2 port can also be shared by multiple subsystems concurrently.

In the figure on this slide, the TCP/IP stacks on System 1 and System 2 are using Queued Direct Input/Output (QDIO) devices for communications.

On System 1, the TCP/IP stack defined an INTERFACE statement with a PORTNAME of OSAE200. When TCP/IP starts the OSAE200 interface, VTAM locates the QDIO TRLE definition statement which has the same defined PORTNAME. VTAM then allocates the READ, WRITE and DATA sub-channels defined for this QDIO device. The QDIO sub-channels are mapped to the actual OSA-Express2 port using CHPID 0A.

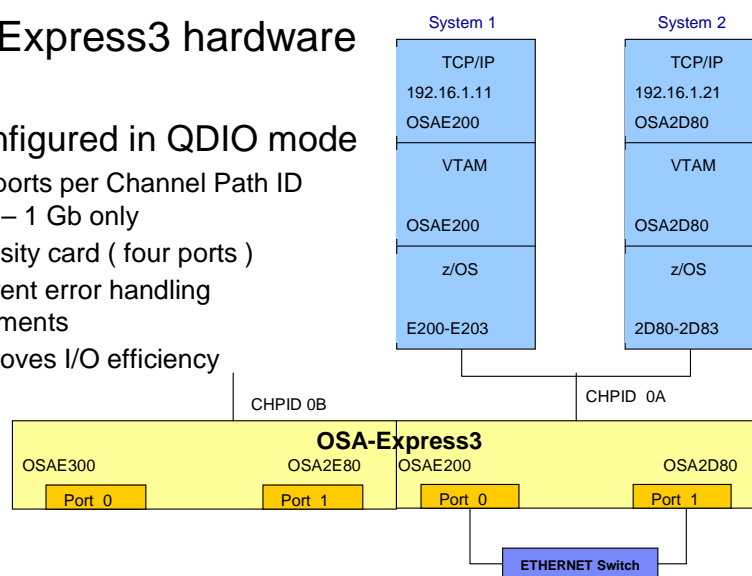
On System 2, the TCP/IP stack defined an INTERFACE statement with a PORTNAME of OSA2D80. When TCP/IP starts the OSA2D80 interface, VTAM locates the QDIO TRLE definition statement which has the same defined PORTNAME. VTAM then allocates the READ, WRITE and DATA sub-channels defined for this QDIO device. The QDIO sub-channels are mapped to the actual OSA-Express2 port using CHPID 0B.

## New OSA-Express3 multiport OSA support

- New OSA-Express3 hardware available

- ▶ When configured in QDIO mode

- Multiple ports per Channel Path ID (CHPID) – 1 Gb only
    - Dual density card ( four ports )
    - Transparent error handling enhancements
      - Improves I/O efficiency



15

Hardware exploitation

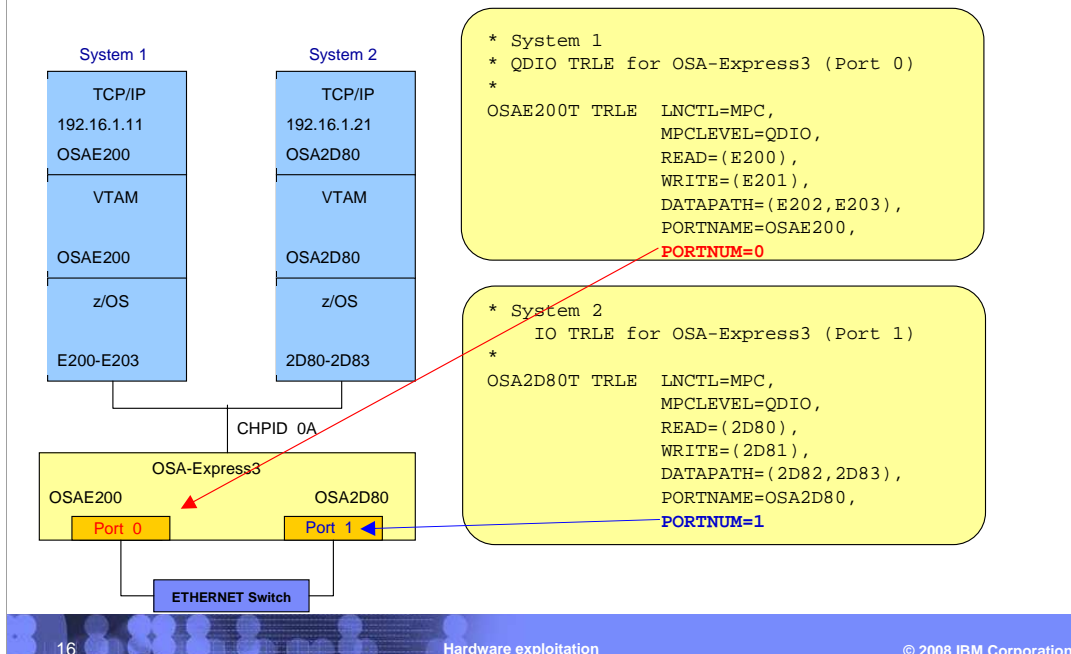
© 2008 IBM Corporation

OSA-Express3 hardware is now available. Some 1Gb OSA-Express3 features configured in Queued Direct Input/Output (QDIO) mode support multiple ports per Channel Path ID (CHPID) in addition to transparent error handling enhancements. QDIO mode has a defined CHPID type of OSD.

The figure on this slide shows that the OSA-Express3 feature is a dual density card which supports 2 CHPIDs, with each CHPID supporting two ports. This support doubles the number of available OSA-Express3 ports with the number of CHPID definitions staying the same. OSA-Express 2 requires two CHPIDs for a feature with two ports; OSA-Express3 requires two CHPIDs for a feature with four ports.

The OSA-Express3 transparent error handling enhancements introduces a higher level of error reporting granularity in the inbound data stream. Before this support, if one packet within the read operation was in error, all packets within that entire read operation were also discarded. Now, OSA-Express3 individually marks packets with errors as not valid. The z/OS Communications Server later discards the packets which are marked as not valid. This new efficiency reduces the number of unnecessary retransmissions which improves I/O efficiency.

## New PORTNUM operand on TRLE definition

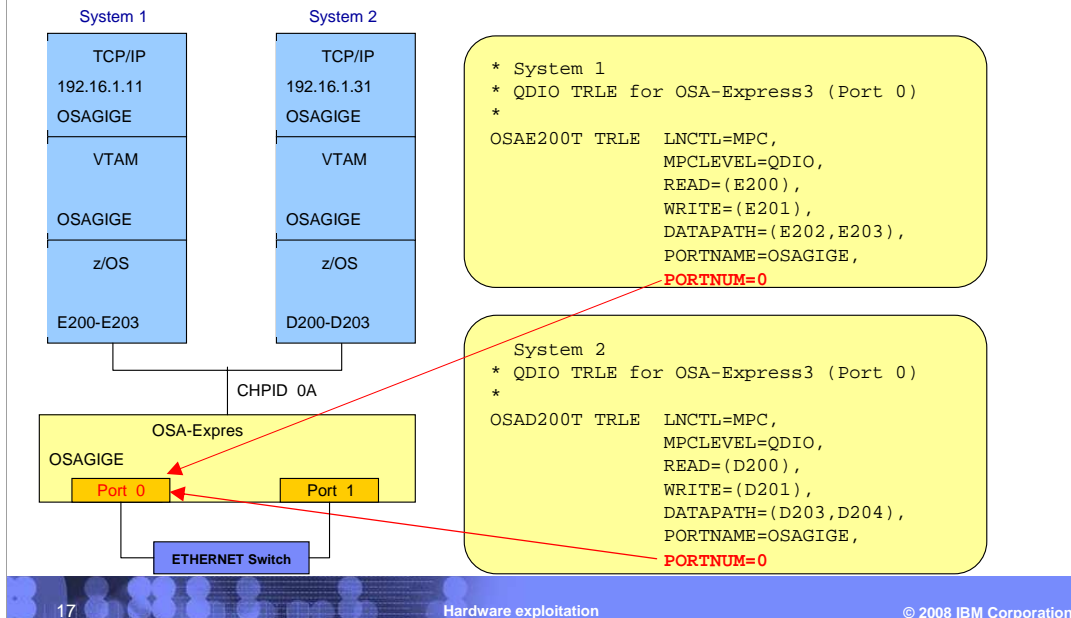


A new PORTNUM operand is provided on the QDIO TRLE definition statement. This new operand allows you to specify which OSA-Express3 port number you want to be associated with your QDIO MPC group. The default is port number 0, or the first physical port available on each CHPID of the OSA-Express3 feature. To support possible future growth, the PORTNUM operand allows a value to support up to sixteen ports, but each OSA-Express3 CHPID supports only two physical ports. If you specify a port number greater than 1, OSA-Express3 will fail to activate and you will see message: **IST1631I OSA2D80T SUBCHANNEL 2D81 INVALID QDIO PORT NUMBER**

Here are sample TRLE definition statements showing the new PORTNUM keyword. This sample shows how you code the new QDIO TRLE definition statement for System 1 and System 2. On System 1, you specify the PORTNUM operand to a value of 0 to access port 0 (CHPID 0A) on the OSA-Express3 feature. On System 2, you specify the PORTNUM operand to a value of 1 to access port 1 on the OSA-Express3 feature.



## OSA-Express3 multiport OSA support shared OSA port example



17

Hardware exploitation

© 2008 IBM Corporation

Here are sample TRLE definition statements showing how two systems share the same OSA-Express3 port. This sample shows how you would code the new QDIO TRLE definition statement for System 1 and System 2. On System 1, you specify the PORTNUM operand to a value of 0 to access port 0 (CHPID 0A) on the OSA-Express3 feature. On System 2, you also specify the PORTNUM operand to a value of 0 to access port 0 (CHPID 0A) on the OSA-Express3 feature.

Notice the two systems define the same PORTNAME of "OSAGIGE" in this example. This is required when sharing a an OSA-Express port.

The TCP/IP definition rules to activate an OSA-Express3 QDIO device remain the same. There are no new changes introduced on the TCP/IP definition statements. The PORTNAME (OSAGIGE in this example) specified on the VTAM TRLE definition statement is still specified as the DEVICE/LINK or INTERFACE name in the TCP/IP profile.

## OSA-Express3 transparent error handling

- OSA-Express3 can invalidate just the packet in error instead of having to invalidate all packets in the read operation
- Sample VTAM Tuning Statistics output

▶ F NET, TNSTAT, TRLE=OSA2D80T, CNSL, TIME=5

```

IST1233I DEV      = 2D82      DIR      = READ
IST1719I PCIREALO =          0 PCIREAL  =        22
IST1720I PCIVIRTO =          0 PCIVIRT  =         0
IST1750I PCIHRSO  =          0 PCIHRSH  =         0
IST1751I PCIUNPRO =          0 PCIUNPRD =         0
IST1752I RPROCDEO =          0 RPROCDEF =         0
IST1753I RREPLDEO =          0 RREPLDEF =         0
IST1754I NOREADSO =          0 NOREADS  =         1
IST1721I SBALCNTO =          0 SBALCNT  =        22
IST1722I PACKCNTO =          0 PACKCNT  =        22
IST2185I FRINVCTO =          0 FRINVCT  =         1
IST1236I BYTECNTO =          0 BYTECNT  =       2796
IST1810I PKTIQDO =          0 PKTIQD  =         0
IST1811I BYTIQDO =          0 BYTIQD  =         0

```

The “transparent error handling” support improves the overall Input/Output efficiency by invalidating only select inbound packets. Previously, if one packet was found to be in error, all packets within that same read operation were considered in error and discarded. Now, if packets are found to be in error by OSA-Express3, only the individual packets are marked as being not valid. The z/OS Communications Server will then discard the invalid packets and process all other valid packets.

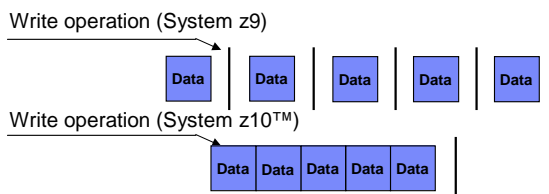
The QDIO tuning statistics output has been modified to add new message IST2185I. This new message only applies to inbound data received on a QDIO DATAPATH device. This new message contains a counter (frinvct) and overflow counter (frinvcto) which represent the accumulated number of packets that were marked as not valid by OSA-Express3.

The counters in message IST2185I will both be zero for OSA-Express and OSA-Express2 QDIO ports.

The sample tuning statistics output above shows the “READ” output for a QDIO DATAPATH device. New message IST2185I will display the number of frames marked as invalid by OSA-Express3 for this DATAPATH device. In this example, only one frame was marked as being invalid

## IBM System z10 EC™ HiperSockets zIIP assisted multiple write facility

- HiperSockets can now move multiple output data buffers in one write operation
  - ▶ Reduces processor utilization
- In z/OS V1R10, this multiwrite operation can be offloaded to a zIIP
  - ▶ Further savings!
  - ▶ Only for TCP traffic that originates in this host
    - No sysplex distributor or EE traffic



HiperSockets is a technology that provides high-speed internal TCP/IP connectivity between logical partitions within a System z®.

IBM System z10 Integrated Information Processor (zIIP) is a specialty processor designed to free up general computing capacity and lower software costs for select workloads.

The IBM System z10 Enterprise Class (EC) includes a new function, HiperSockets Multiple Write. This allows multiple data buffers to be moved from one system image to another across HiperSockets with one operation. This can reduce processor utilization.

When enabled, HiperSockets multiple write is used any time a message spans the HiperSockets frame size, thus requiring multiple output buffers to transfer the message. Therefore, it will only be used for larger outbound messages. Spanning multiple output data buffers can be affected by several factors including HiperSocket frame size, application socket send size, TCP send size, and MTU size.

This multiwrite operation can also be offloaded to a zIIP for further savings. The zIIP offload is only supported for TCP traffic that originates in this host. For example, it cannot be used for sysplex distributor or enterprise extender (EE) traffic.

Restriction: HiperSockets multiwrite is not supported when running as a guest in a z/VM environment. Also, zIIP assist is only supported when using the new multiwrite support to move large outbound data on the zIIP.

## zIIP-assisted HiperSockets multiwrite configuration example

```
GLOBALCONFIG IQDMULTIWRITE ZIIP IQDIOMULTIWRITE
```

Turns on multiwrite function for all HiperSocket devices

Allows multiwrite to run on ZIIP

The multiwrite function (not including zIIP) is rolled back to V1R9 in these PTFs: UK37606 UK37607 UA41863

Both HiperSockets multiple write and zIIP-assisted HiperSockets multiple write are disabled by default. They must be enabled with new GLOBALCONFIG parameters.

Notice in the configuration example on this slide that the IQDMULTIWRITE parameter that turns on the function is spelled differently than the IQDIOMULTIWRITE parameter that enables it for ZIIP assist

If the GLOBALCONFIG parameters are changed with the VARY TCPIP,,OBEYFILE command, the new values do not take effect for any active HiperSockets interfaces. For a change in these parameters to take effect for an active HiperSockets interface, you must stop and restart the interface.

The multiwrite function (not including ZIIP assist) is rolled back to V1R9 in PTFs UK37606, UK37607 and UA41863.

## Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

HiperSockets IBM System z System z10 VTAM z/OS z10 EC

A current list of other IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2008. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.