Virtual Ethernet Bridging

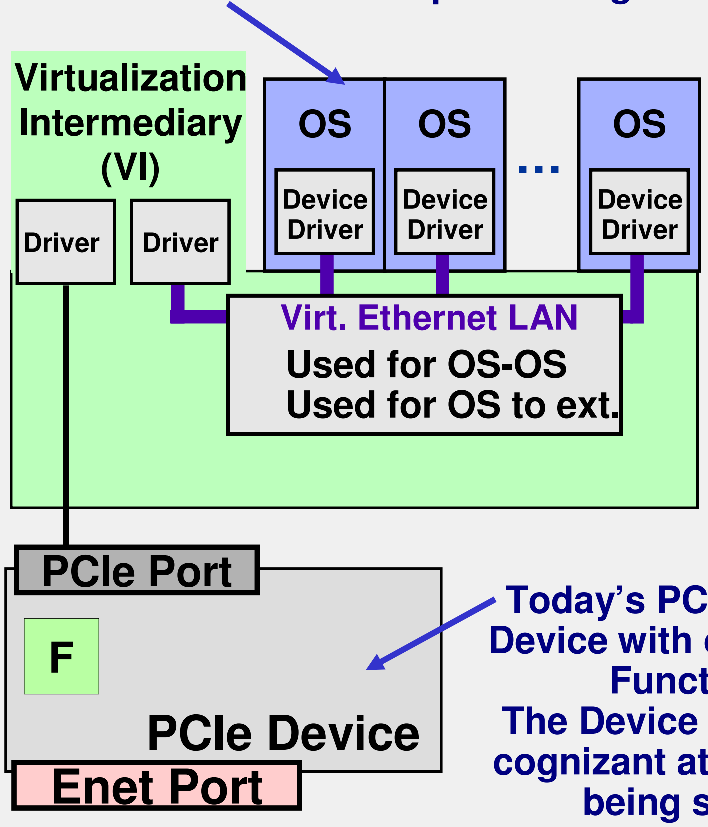
Mike Ko
Renato Recio

IBM

IO Shared Through Virtualization Intermediary (e.g. Hypervisor)

2007 Standard High Volume Server PCI Device Sharing → Example

Operating Systems (a.k.a. guest OS, SIs) share the adapter through a Virtualization Intermediary. All MMIO and DMA operations go through VI.

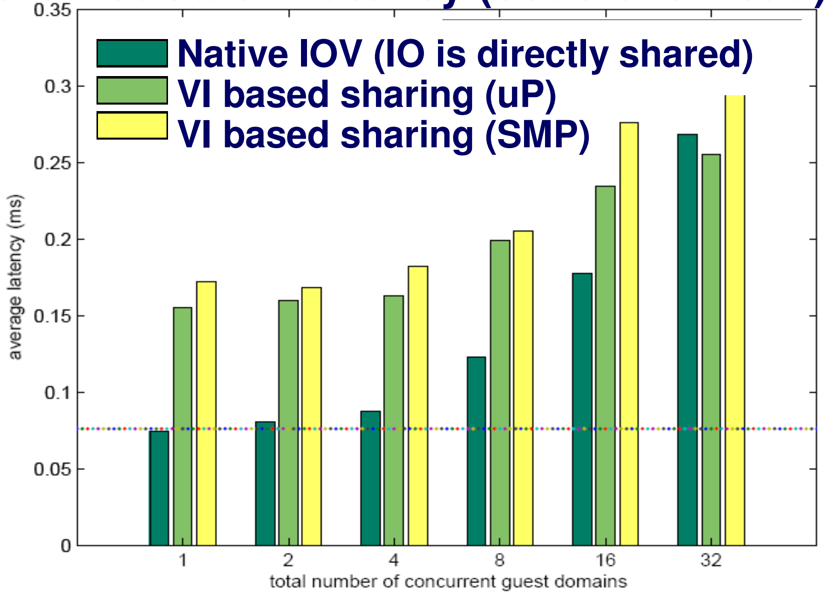


Today's PCIe Ethernet Device with one or more Functions.
The Device may not be cognizant at all that it is being shared.

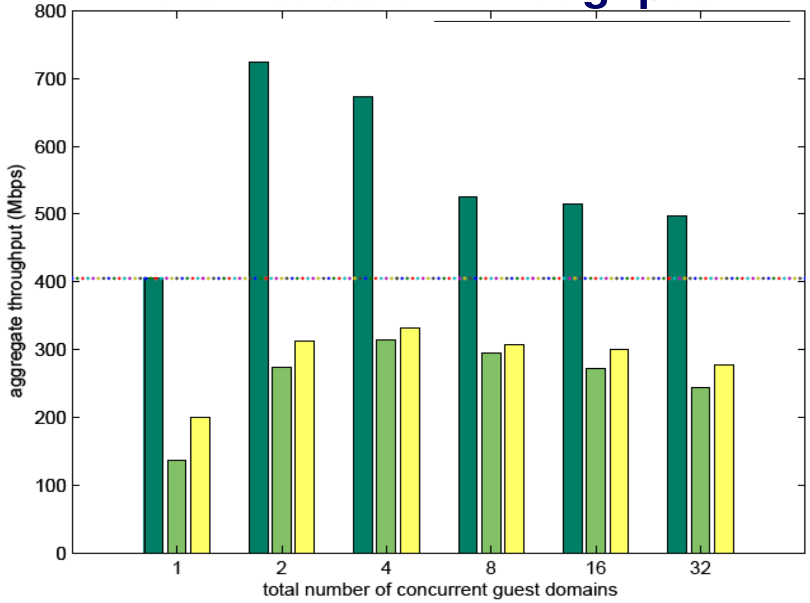
- **Virtualization Intermediaries (VIs) are used to safely share IO.**
  - ► That is, 1 or more OSs share the PCI device through the VI.
  - ► VI may be part of Hypervisor or not.
- **The VI performs Ethernet sharing functions:**
  - ► Multiplexes flows from multiple OSs;
  - ► Performs PCI IO transactions on behalf of the OS;
  - ► Provides a communication mechanism between OSs running above the same Hypervisor;
  - ► …
- **The PCIe Device typically supports:**
  - ► Multi-MAC, to allow a MAC per OS;
  - ► One or more PCIe Functions, with one of more Transmit & Receive Queue Pairs;
  - ► State of the Art IP stack accelerators;
  - ► …

IO Virtualization Trends

Ethernet Stack Latency (code overhead)



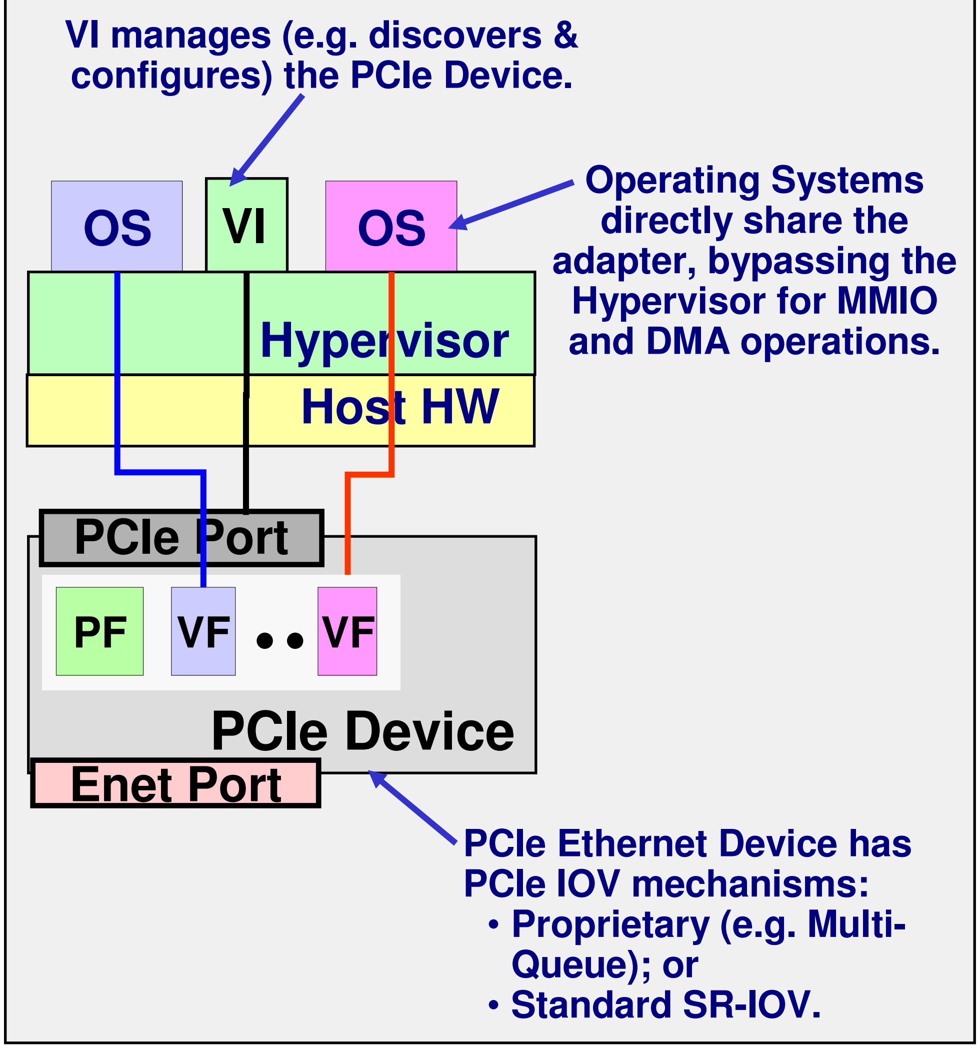
Ethernet Stack Throughput*



- **VI based IOV adds path length on every IO operation.**
- **Native IOV uses direct sharing mechanisms in the PCIe Device to enable "Hypervisor bypass".**
  - ►Significantly improves performance, in example above, Native IOV doubled the throughput and reduced latency by up to half.
- **Native IOV is becoming increasingly important, due to several factors, the primary factors are:**
  - ►IT budget pressures, increasing the demand for Virtualization; and
  - ►More cores per socket, increasing the number of OSs per socket.

Native IOV

Native IO Virtualization Example



VI manages (e.g. discovers & configures) the PCIe Device.

Operating Systems directly share the adapter, bypassing the Hypervisor for MMIO and DMA operations.

PCIe Ethernet Device has PCIe IOV mechanisms:

- Proprietary (e.g. Multi-Queue); or
- Standard SR-IOV.

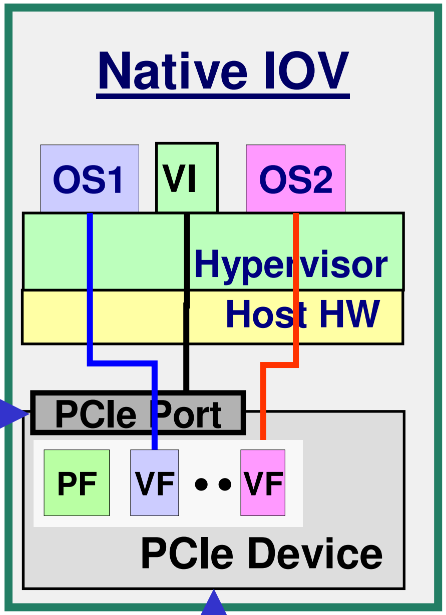
- Native IO Virtualization (IOV) uses direct sharing mechanisms in the PCIe Device to enable “Hypervisor bypass”.
- Two approaches have been use to support Native IOV on PCIe Ethernet Device’s:
  - Multi-queue, vendor proprietary.
  - PCIe Single-Root IO Virtualization, standard for “Northside” (i.e. PCIe Port) Native IOV mechanisms.
- PCIe SR-IOV is being widely adopted by PCIe Ethernet Device Vendors.
- However, for Ethernet, an additional Native IOV mechanism is needed to cover the “Virtual Ethernet Bridge” (VEB) used to communicate between OSs running above the same Hypervisor.

Native IOV… part 2

PCI SIG SR-IOV only standardized "North Side" interface

Device is directly shared.

- Each OS is assigned a Virtual Function (VF).
- Each VF has 1 or more Queue Pairs (QPs).
- QPs are used to communicate directly with adapter.



SR-IOV moves the Hypervisor's "Virtual Switch" out of the Hypervisor.

(often called "Virtual Ethernet Bridge", VEB, by networking vendors)

No standard exists for VEB.

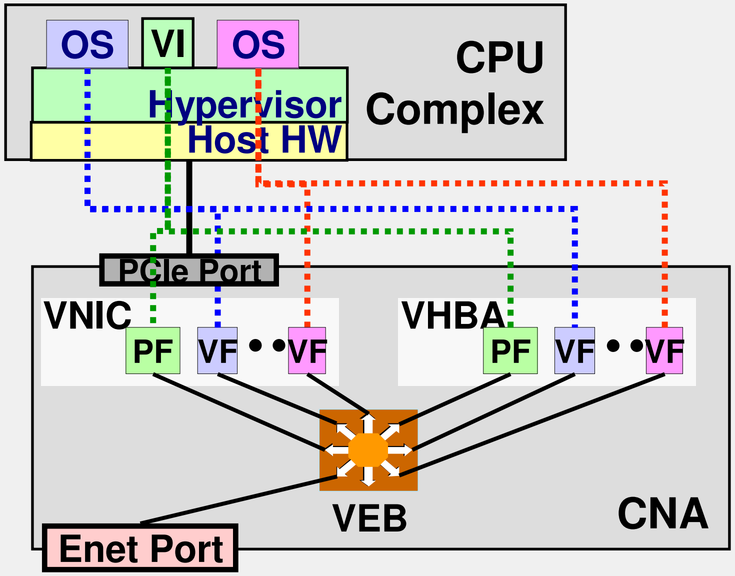
Why is a common VEB definition important?
Because there are no mechanisms to uniquely identify OSes

- Nothing to prevent OS2 from taking over OS1's personality

Robust Access Control and QoS mechanisms are needed for virtual servers attached to converged fabrics

VEB Approaches

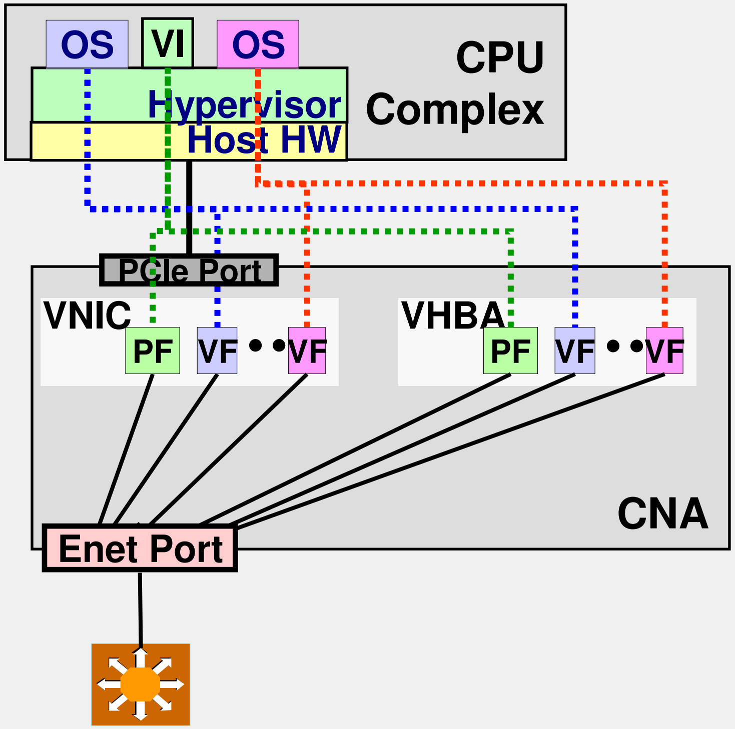
VEB in Adapter



As covered before, done by most vendors today.

However, to enable wide adoption (i.e. minimize Hypervisor, VI and OS impact), requires commonality (see next page).

VEB in Switch

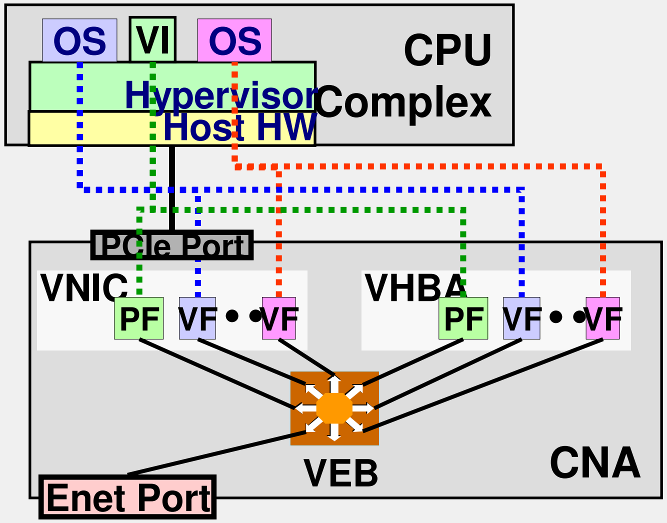


Not done by switch vendors today.

A new routing mechanism would be used to enable VEB, while providing the necessary port ACLs.

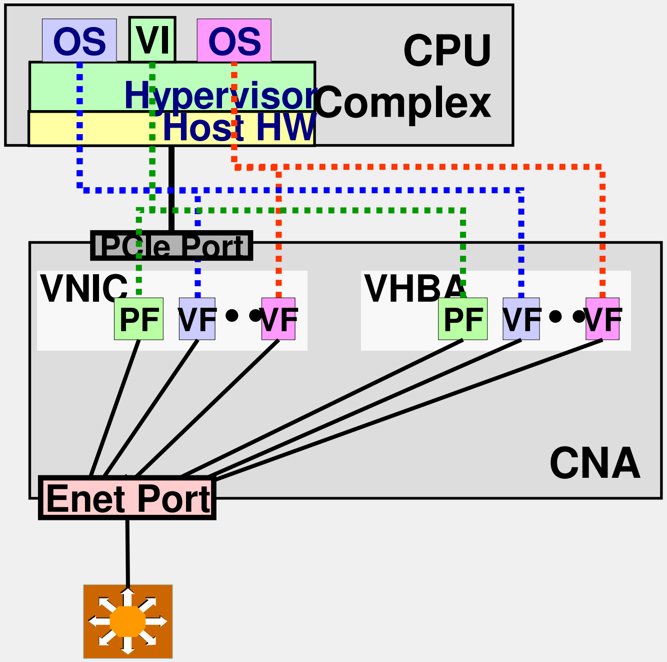
Pros/Cons Approaches

VEB in Adapter



- **Pros**
  - ► Higher bandwidth (PCIe level)
  - ► Lower latency (no external, 2 us switch)
  - ► Standardizes PCI VEB semantics
- **Cons**
  - ► PCI vendor VEB semantic differences.
  - ► Does not leverage vendor ACLs.

VEB in Switch



- **Pros**
  - ► Leverages vendor ACLs.
- **Cons**
  - ► Lower bandwidth (data goes thru Enet port)
  - ► Higher latency

Observations

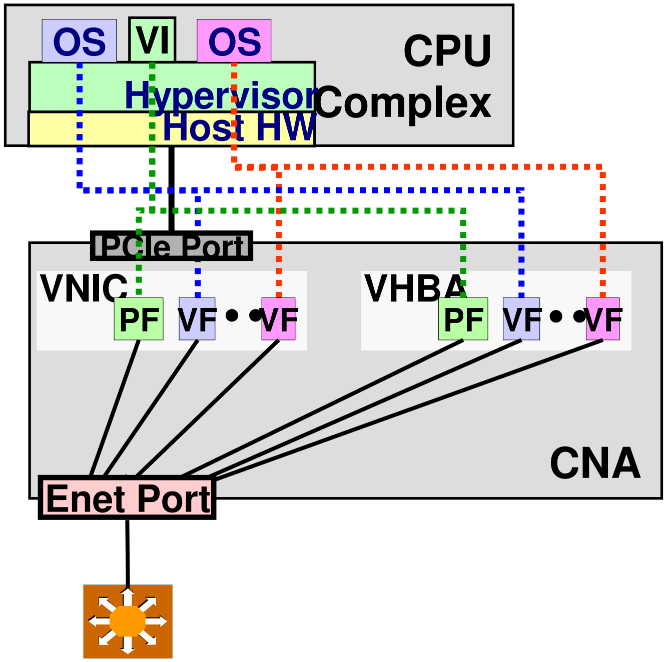
- **Adapter vendors are offering basic VEB in adapter approach today, but lack:**
  - ► Robust Access Control and QoS capabilities;
  - ► Common function set; and
  - ► Common interface (syntax and semantics) for the functions.

- **Networking IHVs may pursue a VEB in switch approach.**
  - ► If so, wire protocol would be standardized through IEEE 802.

- **In our view, both (VEB in adapter & VEB in switch) approaches will co-exist.**
  - ► VEB in Adapter can be done without wire new Ethernet protocols.
  - ► VEB in Switch will require new wire protocols.

- **Why does IEEE need to do anything for “VEB in Switch”?**

Scope of Requirement Analysis Associated with VEB in Switch

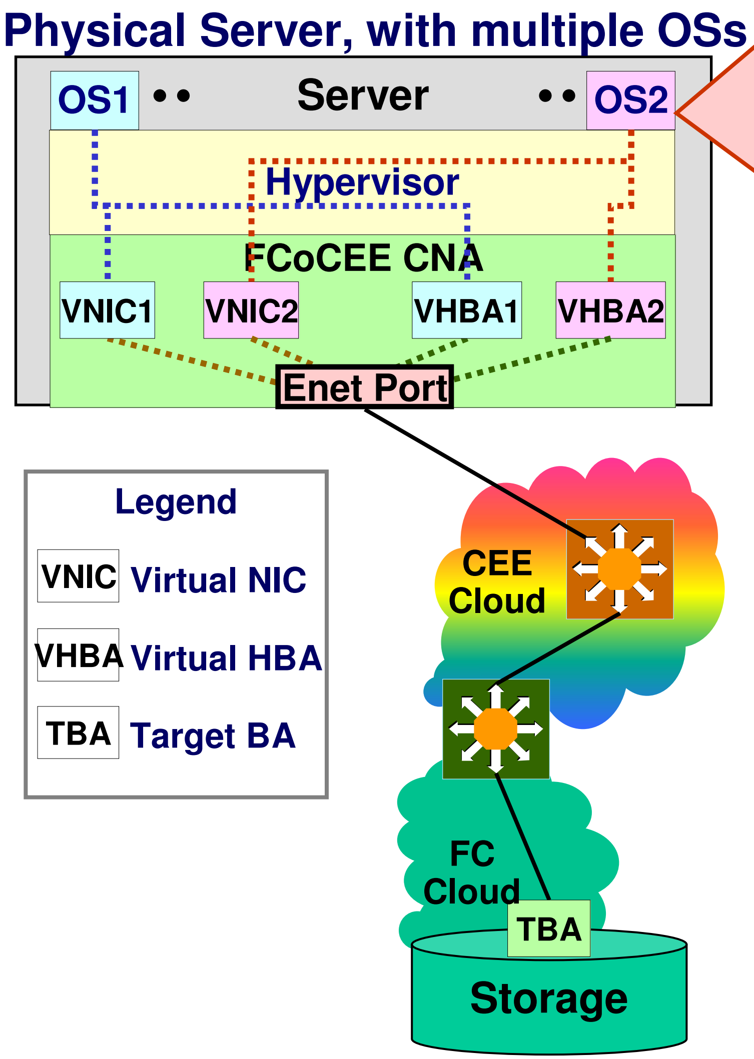


- **Ethernet Devices are coming to market:**
  - With SR-IOV Version 1 support:
    - Minimally one PF per Ethernet port
    - Minimally one or more VF per PF
  - Several forms of Ethernet convergence:
    - iSCSI or iSCSI over DCB
    - FCoE or FC over DCB

- **As mentioned previously, for these devices the Hypervisor's VEB mechanism will be "outboarded".**

- **For the "VEB in Switch" approach:**
  - Unicast access controls mechanisms will be needed to assure one OS doesn't assume another OS's personality.
  - Multicast/Broadcast controls also be needed.
  - Port Mirroring/Routing mechanisms will be needed to allow Intrusion Detection & Prevention to run in a virtual OS.
  - VLAN mechanisms may need to be automated.

ACL Annex (Alone) Does Not Covers Virtual Fabric



Today's IEEE protocols do not protect a Server with 1000s (e.g. 64,000) Virtual OSs from the following attacks.

- **Attack 1: OS2 VNIC2 can send Ethernet packets using OS1's VHBA1 MAC**
- **Attack 2: OS2 VNIC2 can send Ethernet packets using OS1's VHBA1 MAC, a target assigned to OS1's NPID, etc…**

Proposal Going Forward

- **For “VEB in Adapter”:**
  - ► IBM is recommending to PCI SIG that the PCI IOV WG analyze the requirements.

- **For “VEB in Switch”:**
  - ► IBM recommends the IEEE 802 define the requirements and,
    due to PCIe SR-IOV schedules, quickly create a new PAR for this effort.

  - ► IBM recommends companies work together to define proposal for
    “VEB in Switch” requirements and the associated PAR.