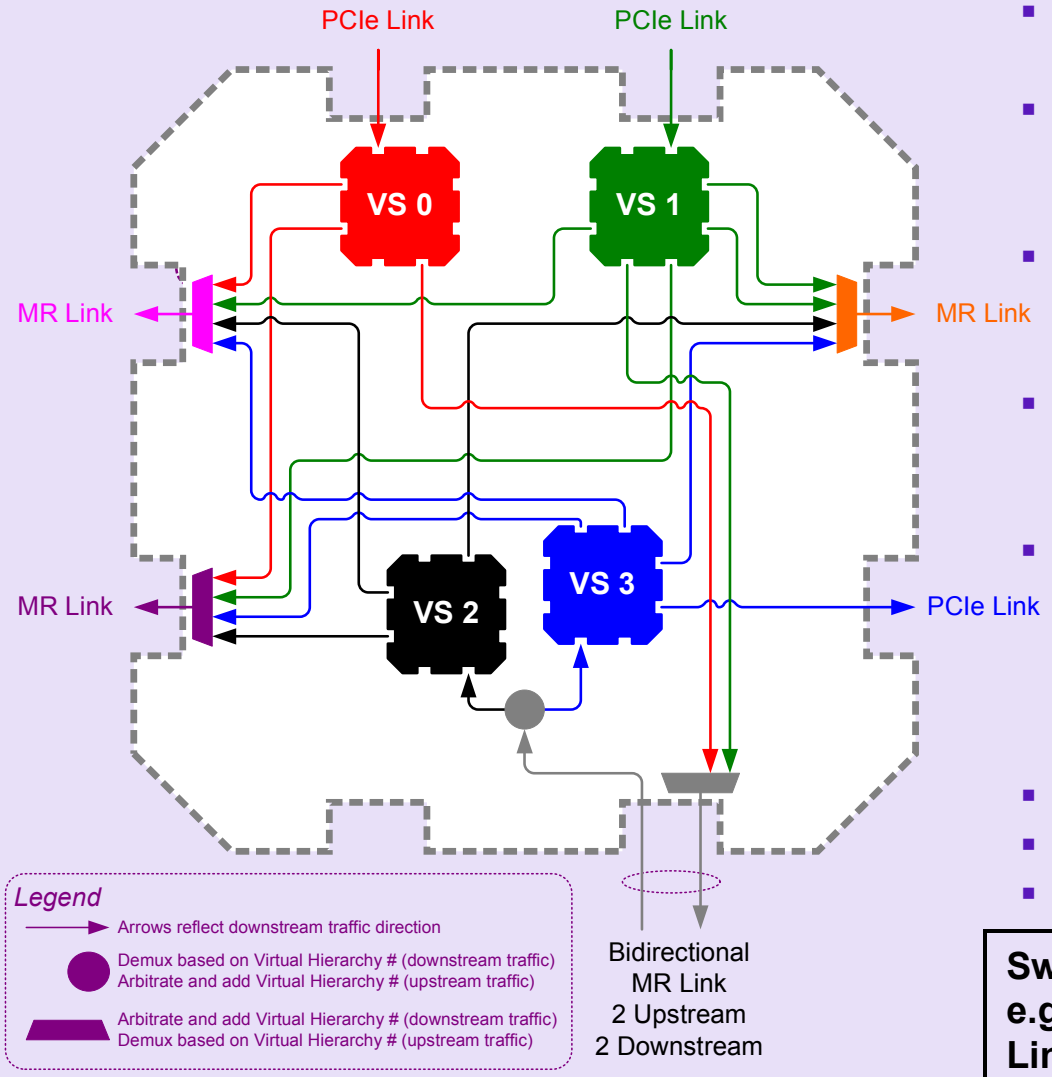

PCI
SIG®

Multi-Root Initialization and Config Space

Steve Glaser (NextIO)

MR Switch Structure

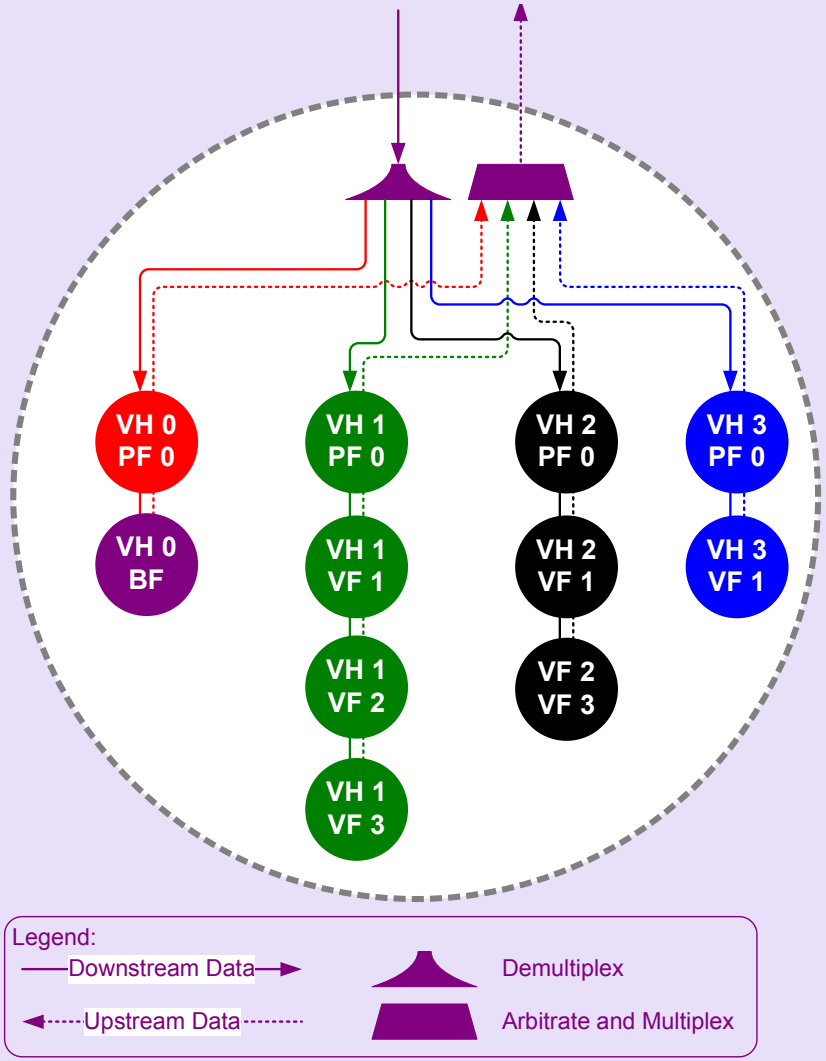


- MR Switch with 4 Virtual Switches
  - ✓ Each Virtual Switch is a PCIe Switch
- VS 0 (Red)
  - ✓ PCIe Link Upstream
  - ✓ 3 MR Links Downstream
- VS 1 (Green)
  - ✓ PCIe Link Upstream
  - ✓ 5 MR Links Downstream
- VS 2 (Black)
  - ✓ MR Link Upstream
  - ✓ 3 MR Links Downstream
- VS 3 (Blue)
  - ✓ MR Link Upstream
  - ✓ 3 MR Links Downstream
  - ✓ 1 PCIe Link Downstream

- MR Links arbitrate between TLPs for each VS
- MR Links demultiplex TLPs to each VS
- One or more VS *Authorized* to Manage Switch

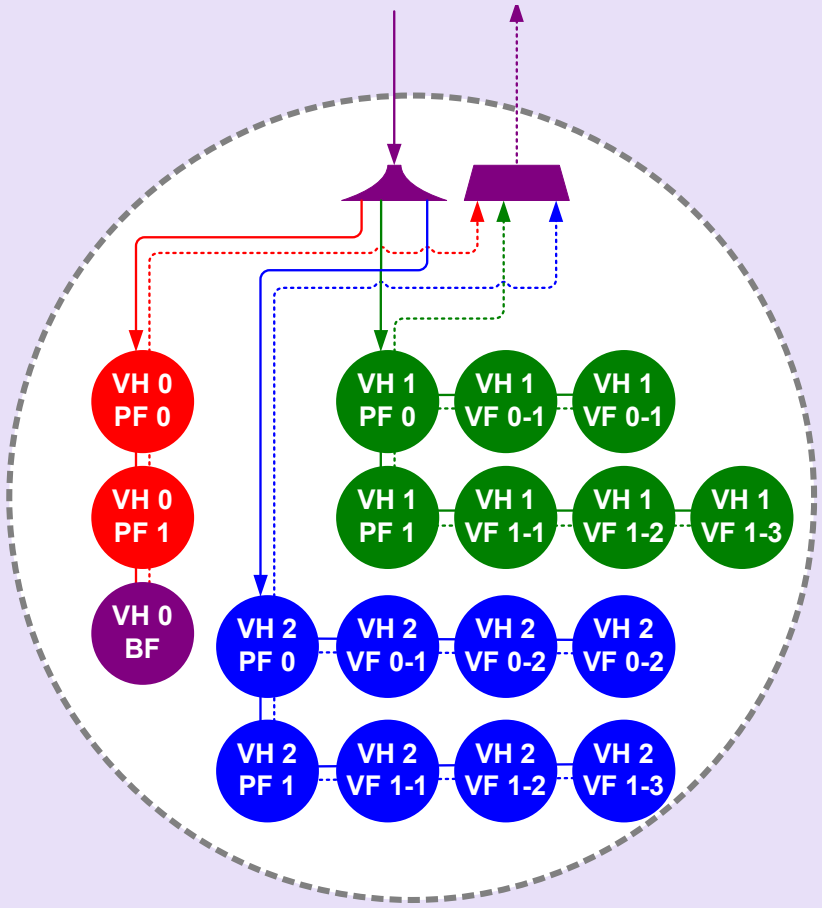
Switch Mapping involves the "white" stuff: e.g. "wires" connecting Virtual Switches to Links as well as arbiters and demultiplexors

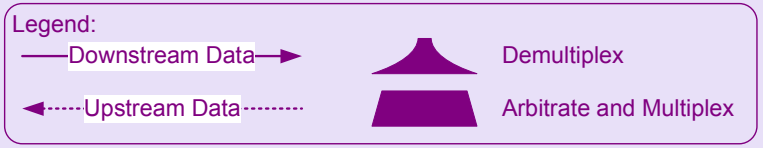
MR Device Structure



- MR Device with 4 Virtual Hierarchies
  - ✓ Single Function Device
- Each VH has:
  - ✓ One Base Function (BF) or
  - ✓ One or more PFs or
  - ✓ One BF and one or more PFs
- BF used to manage the Device
  - ✓ One in every Authorized VH

Two Function MR Device



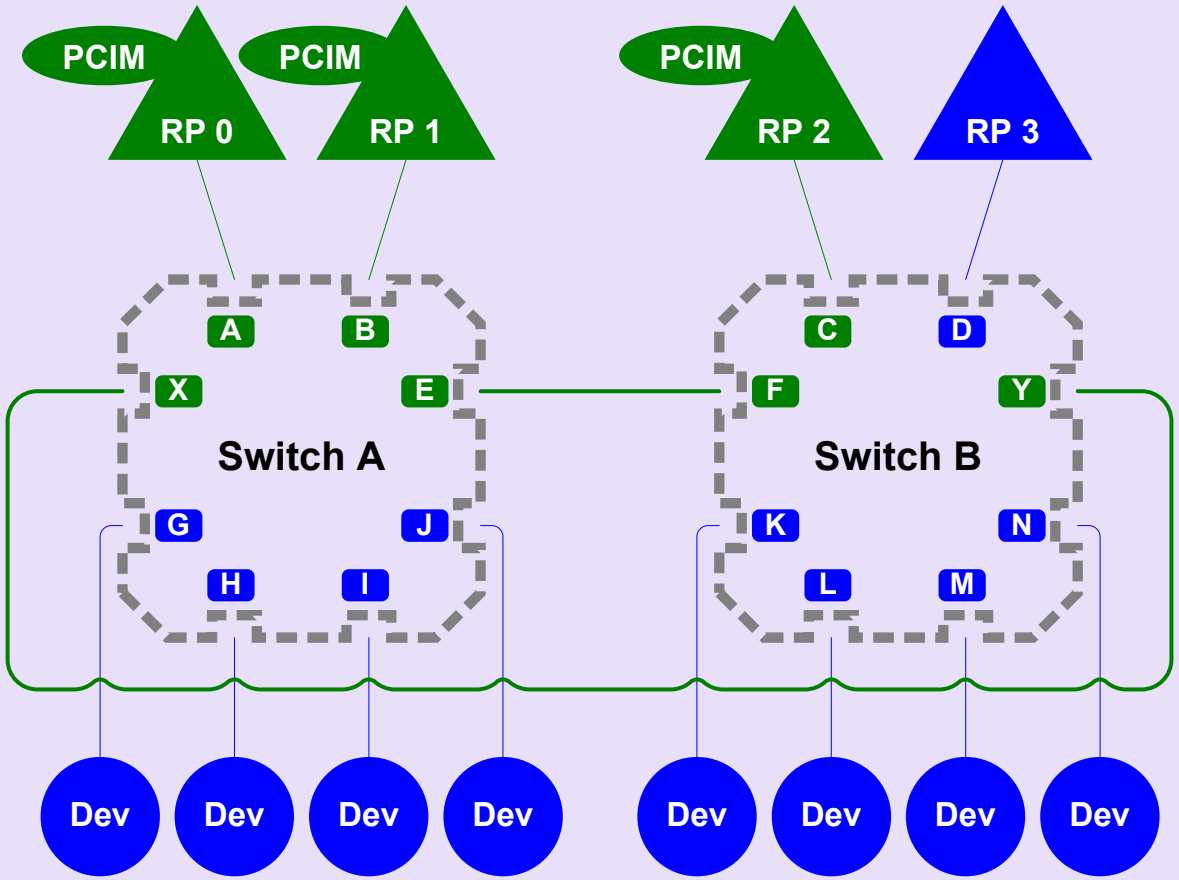


- MR Device with 3 VHs
- One BF shown (VH 0)
  - ✓ VH 1 and VF 2 could have BF
- Each PF may have VFs (SR-IOV)
  - ✓ VF counts *need not* be the same

MR Switch *Initial* Port Types

- PCIM Capable Port:
  - ✓ Link Direction is Upstream
  - ✓ VH 0 is Upstream Port in some VS
    - – VH 1..N mapping Vendor Specific
  - ✓ VH 0 Upstream P2P Bridge Config Header has MR-IOV Capability
  - ✓ VH 0 Upstream VS is Authorized
- Non-PCIM Capable Port:
  - ✓ Link Direction is Vendor Specific
  - ✓ VH 0 not Upstream Port of any Authorized VS
    - – Mapping otherwise Vendor Specific
    - – Any VH could be mapped to
      - Nothing
      - Any downstream bridge
      - Non-Authorized VS upstream bridge
  - ✓ Need not have MR-IOV Capability
    - – If present, subset functionality
- Non-PCIe Management Port:
  - ✓ Vendor Specific, could use I^2C, USB, Ethernet, …
  - ✓ Looks like PCIM Capable PCIe port
    - – Upstream Port of Authorized VS with MR-IOV Capability, …
    - – Issues and responds to subset of TLPs:
      - Config read/write, Memory read/write, MSI write, Fault / PME messages, …
  - ✓ One port can manage entire MR fabric

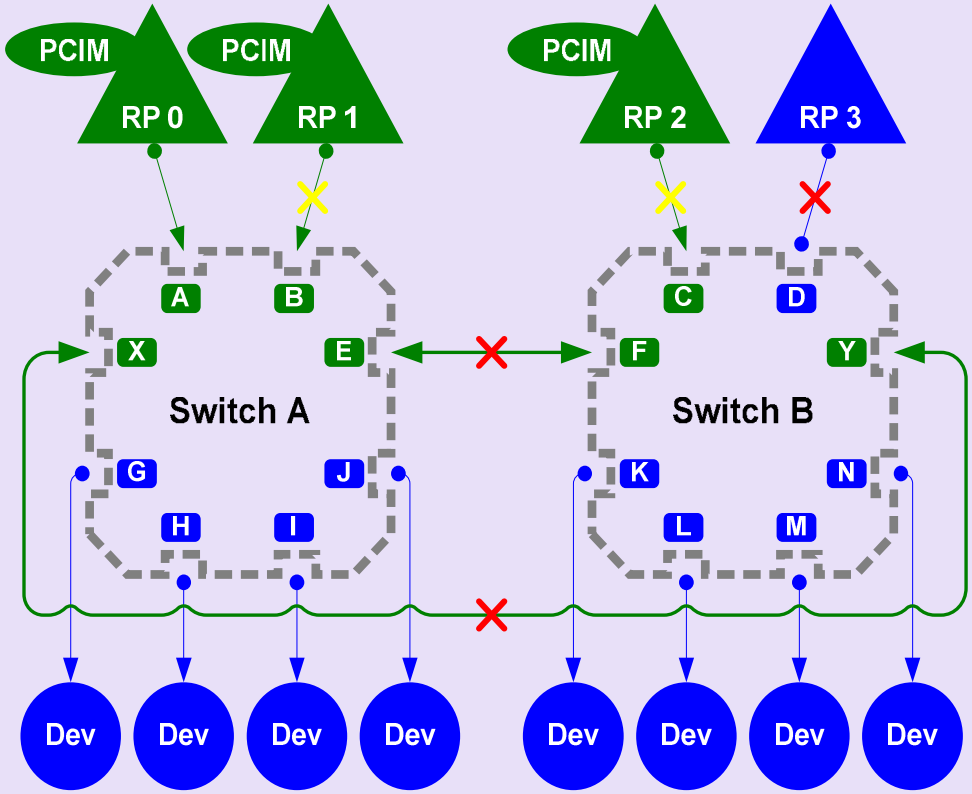
System Picture



Initialization Phases

1. Initial State after Reset
2. PCIM Location Policy Decision
   - ✓ Policy: Out of Scope
3. Topology Discovery
   - ✓ Determine what links exist and how they are connected
4. Component Discovery
   - ✓ Examine MR component capabilities
5. Mapping Policy Decision
   - ✓ Policy: Out of Scope
6. Mapping Implementation
   - ✓ Configure MR components to implement Mapping Policy
7. Virtual Hierarchy Enumeration
   - ✓ Traditional PCIe Enumeration but within a VH

System Picture: Initial State after Reset

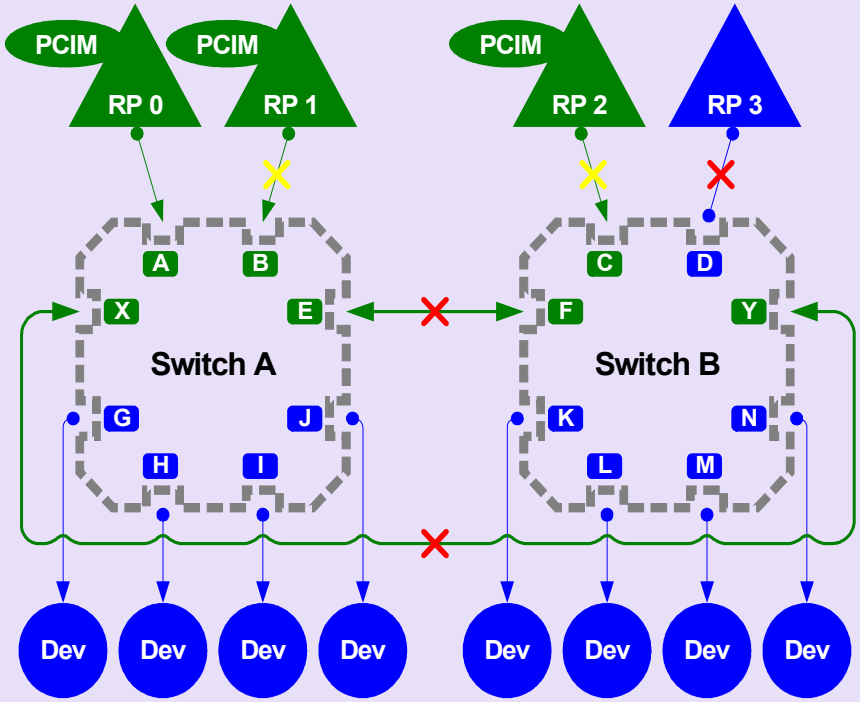


- PCIM Capable Ports
  - ✓ A, B, C
  - ✓ E, F, X, Y
- Non-PCIM Ports
  - ✓ D
  - ✓ G, H, I, J
  - ✓ K, L, M, N
- Links That Don't Train
  - ✓ Downstream ⇔ Down
    - D ⇔ RP3
  - ✓ Upstream ⇔ Up
    - E ⇔ F
    - X ⇔ Y

Policy: Initial PCIM Location

- Assume PCIM in RP 0
- RP 1, RP 2 Held Off
  - ✓ How: Out of Scope

Topology Discovery: (1 of 2)

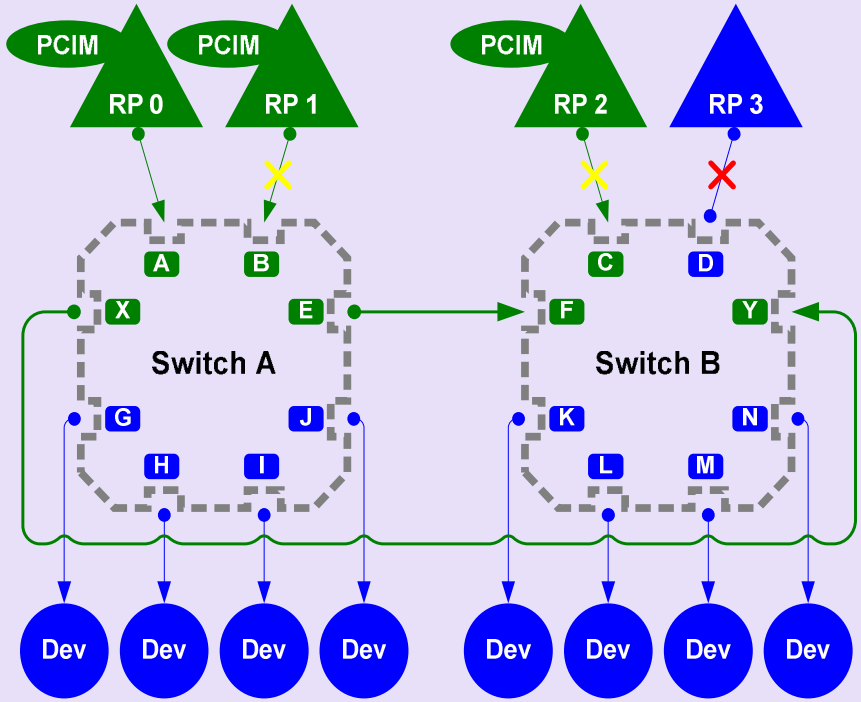


MR-PCIM in RP0:

1. Reads Config Header at Port A
   - Detects Type 1 with MR-IOV Capability
   - Detects that Port A, VH 0 was mapped to VS n
2. Assigns MR Switch number 42 to A
3. Notices Port E detected but didn't train
   - Switches E to downstream allowing E ⇔ F to train
4. Notices Port X detected but didn't train
   - Switches X to downstream allowing X ⇔ Y to train
5. Notices Ports G, H, I, J, E and X trained as downstream
   - Maps VH 0 into downstream P2P Bridges of VS n
   - Optional: Vendor Specific initial mapping might have already done so
6. Queries Devices connected to G, H, I & J
   - Since these are not MR Switches → no additional discovery needed

… continued on next slide

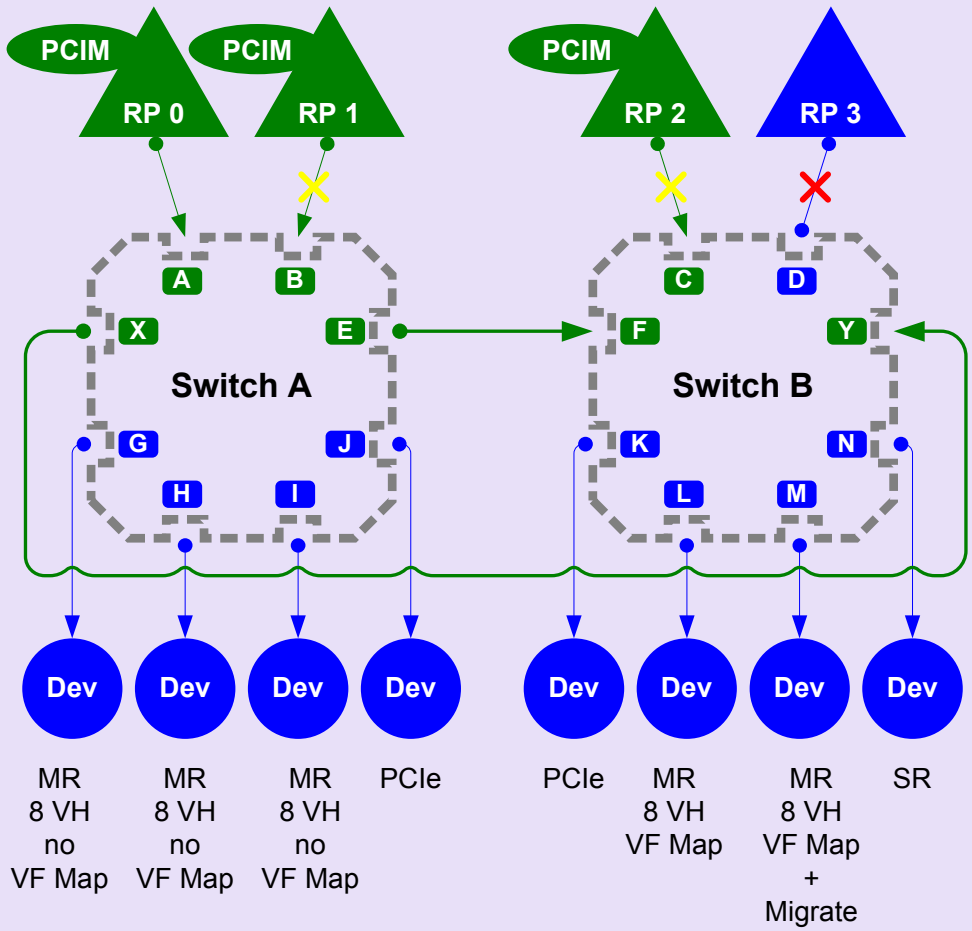
Topology Discovery: (2 of 2)



continued from previous slide …

7. Reads Config Header at Port F
   - Detects Type 1 with MR-IOV Capability
   - Detects that Port F, VH 0 was mapped to VS m.
8. Assigns MR Switch B switch number 86
9. Notices Ports K, L, M and N trained downstream
   - Maps VH 0 into downstream P2P Bridges of VS m
   - Optional: Vendor Specific initial mapping might have already done so
10. Reads Devices connected to K, L, M & N
    - Since these are not MR Switches → no additional discovery needed
11. Notices Port D detected but didn't train
    - Switch D to upstream allowing D ⇔ RP 3 to train
    - Optional: Defer to keep RP 3 from enumerating
12. Reads Config Header at Port Y via Port X
    - Detects Type 1 with MR-IOV Capability
    - Detects MR Switch number of 86
    - Link X⇔Y is a second path to Switch B → no additional discovery needed

Component Discovery:

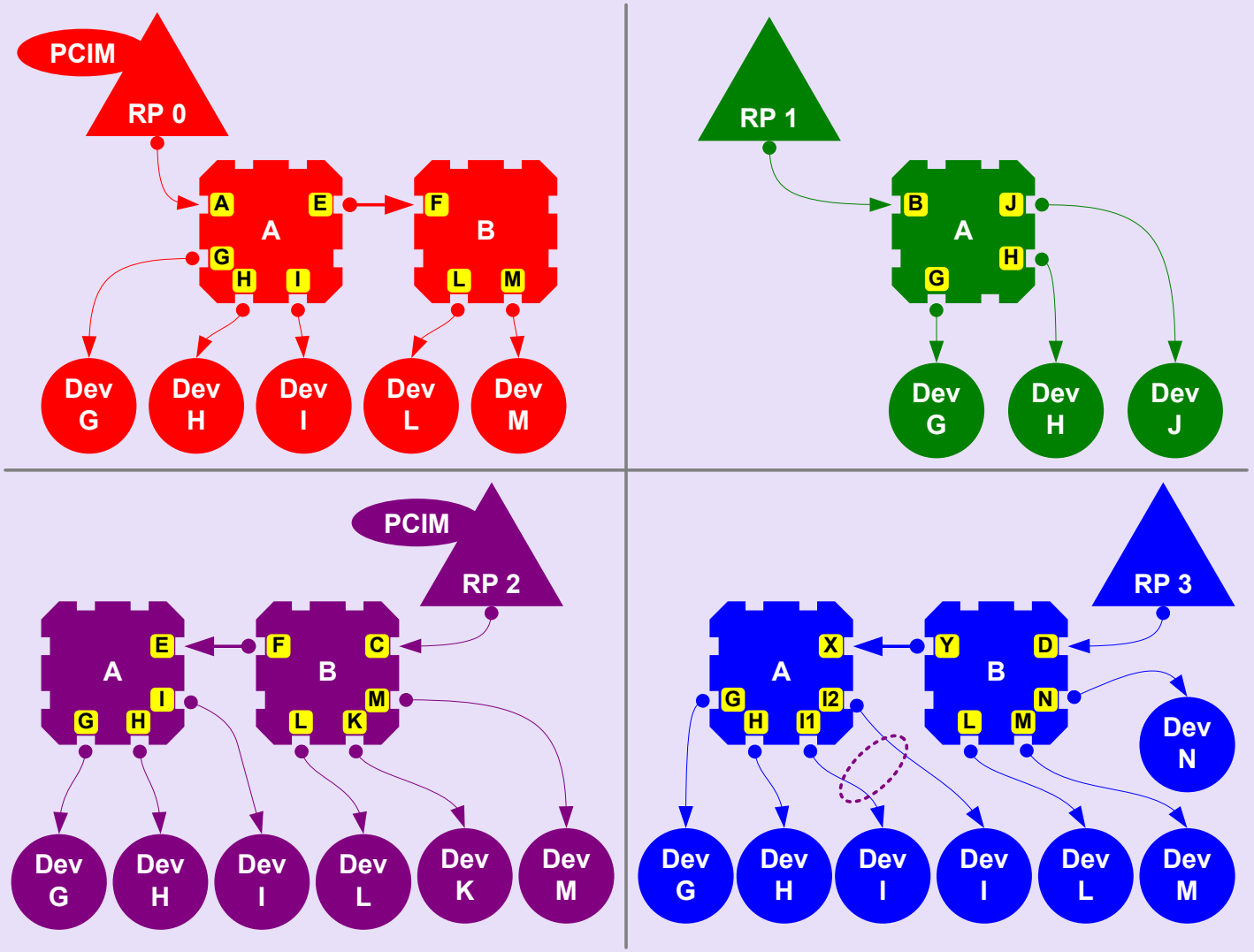


- Device Type 0 Config Space
  - ✓ MR Devices at G, H and I
    - – Single Function Devices
    - – Devices support 8 VHs
    - – No VF Mapping
  - ✓ PCIe Devices J and K
  - ✓ MR Devices L and M
    - – Single Function Devices
    - – Devices support 8 VHs
    - – L: 32 LVFs, 32 MVFs, Mapping only
    - – M: 32 LVFs, 28 MVFs, Mapping and Migration
  - ✓ SR Device N
- Switch Type 1 Config Space
  - ✓ MR Switches A and B
    - – Each supports 8 VSs
    - – Each port supports 8 VHs
- Root Ports
  - ✓ PCIe Protocol
    - – How: Out of Scope

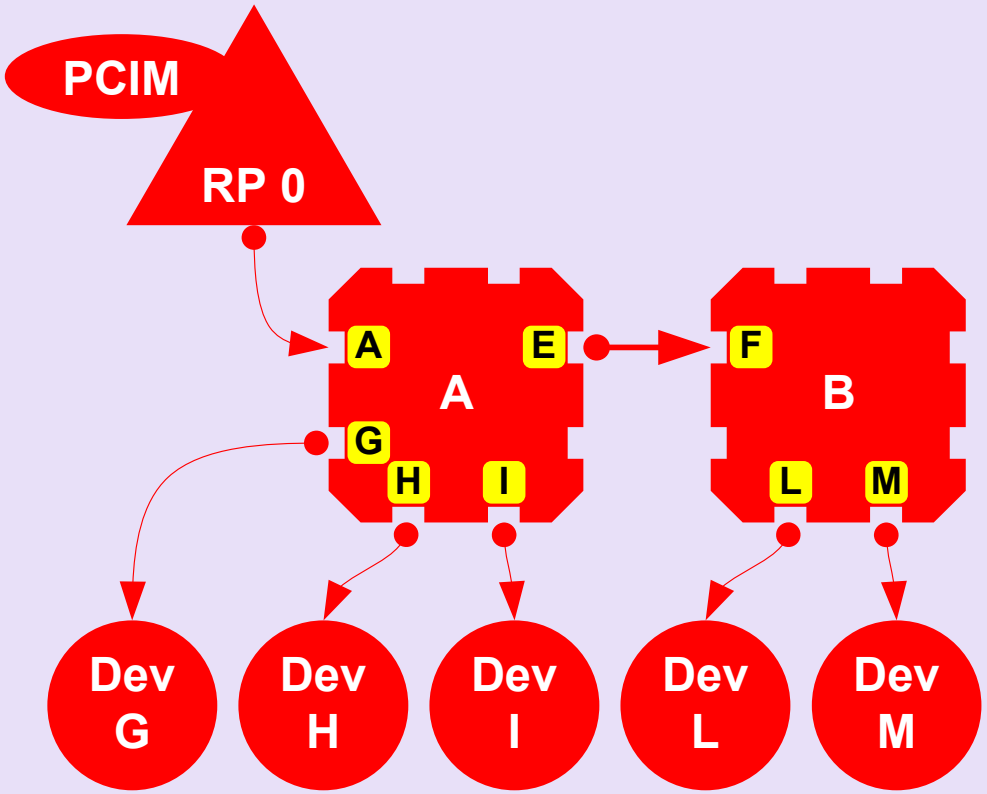
Policy: VH and VF Mapping

VH	Authorized	VH Mapping	VF Mapping
RP 0	**Yes** (MR-PCIM)	VS in Switch A and B PF in Devices G, H, I, L and M	Device L: PF + 15 VFs Device M: PF + 7 VFs (2 unpopulated)
RP 1	No	VS in Switch A PF in Devices G and H Device J	
RP 2	**Yes** (Backup MR-PCIM)	VS in Switch A and B PF in Devices G, H, I, L and M Device K	Device L: PF + 7 VFs Device M: PF + 7 VFs (2 unpopulated)
RP 3	No	VS in Switch A and B PF in Devices G, H, L and M Two PFs in Device I Device N	Device L: PF + 7 VFs Device M: PF + 15 VFs (2 unpopulated)

VH Mapping Policy (Overview)

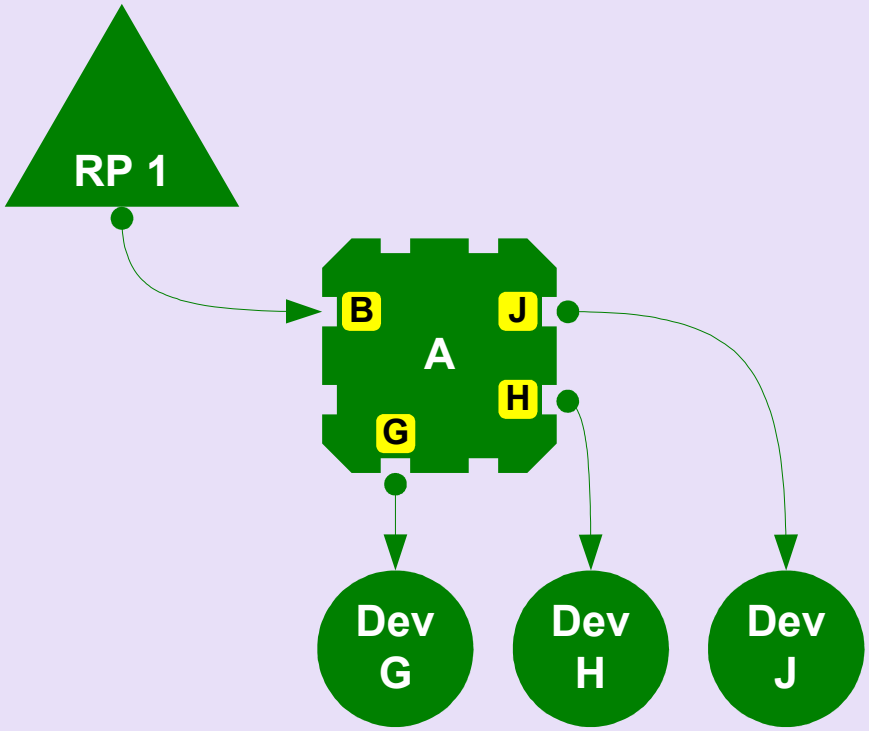


VH Mapping Implementation – RP 0



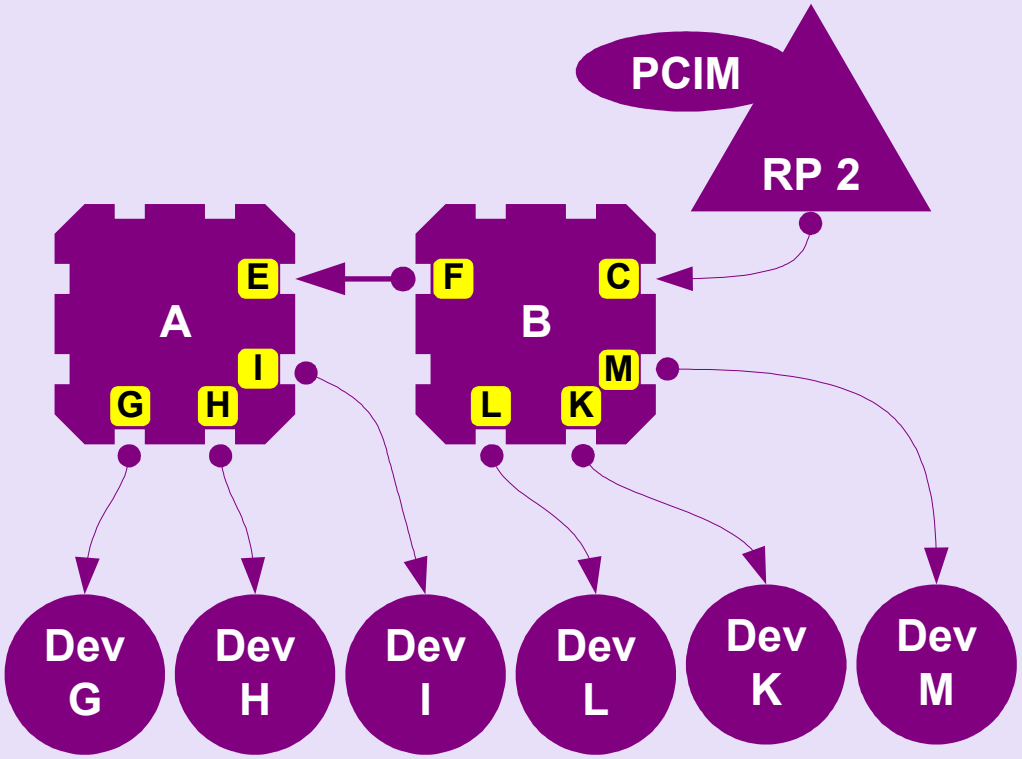
- Switch A: VS 0 Authorized
  - ✓ Upstream: Port A VH0 (from RP 0)
  - ✓ Downstream 0: Port G VH 0
  - ✓ Downstream 1: Port H VH 0
  - ✓ Downstream 2: Port I VH 0
  - ✓ Downstream 3: Port E VH 0 (to SW B)
- Switch B: VS 1 Authorized
  - ✓ Upstream: Port F VH 0 (from SW A)
  - ✓ Downstream 0: Port L: VH 1
  - ✓ Downstream 1: Port M: VH 1
  - ✓ Downstream 2: —
  - ✓ Downstream 3: —

VH Mapping Implementation – RP 1



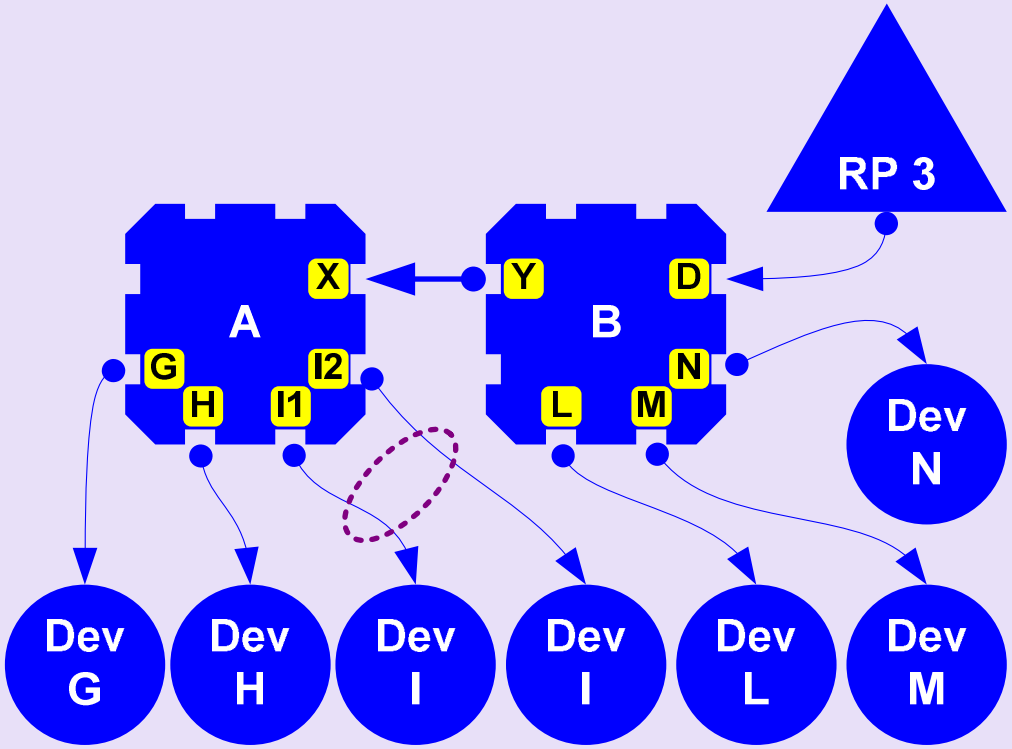
- Switch A: VS 2
  - ✓ Upstream: Port B PCIe (from RP 1)
  - ✓ Downstream 0: Port G VH 2
  - ✓ Downstream 1: —
  - ✓ Downstream 2: Port H VH 1
  - ✓ Downstream 3: Port J PCIe
  - ✓ Downstream 4: —
  - ✓ Downstream 5: —
  - ✓ Downstream 6: —
  - ✓ Downstream 7: —

VH Mapping Implementation – RP 2



- Switch B: VS 0 Authorized
  - ✓ Upstream: Port C PCIe (from RP 2)
  - ✓ Downstream 0: Port F VH 1 (to SW A)
  - ✓ Downstream 1: Port L VH 0
  - ✓ Downstream 2: Port M VH 0
  - ✓ Downstream 3: Port K PCIe
- Switch A: VS 2 Authorized
  - ✓ Upstream: Port E VH 1 (from SW B)
  - ✓ Downstream 0: Port G VH 1
  - ✓ Downstream 1: —
  - ✓ Downstream 2: Port H VH 2
  - ✓ Downstream 3: Port I VH 1

VH Mapping Implementation – RP 3



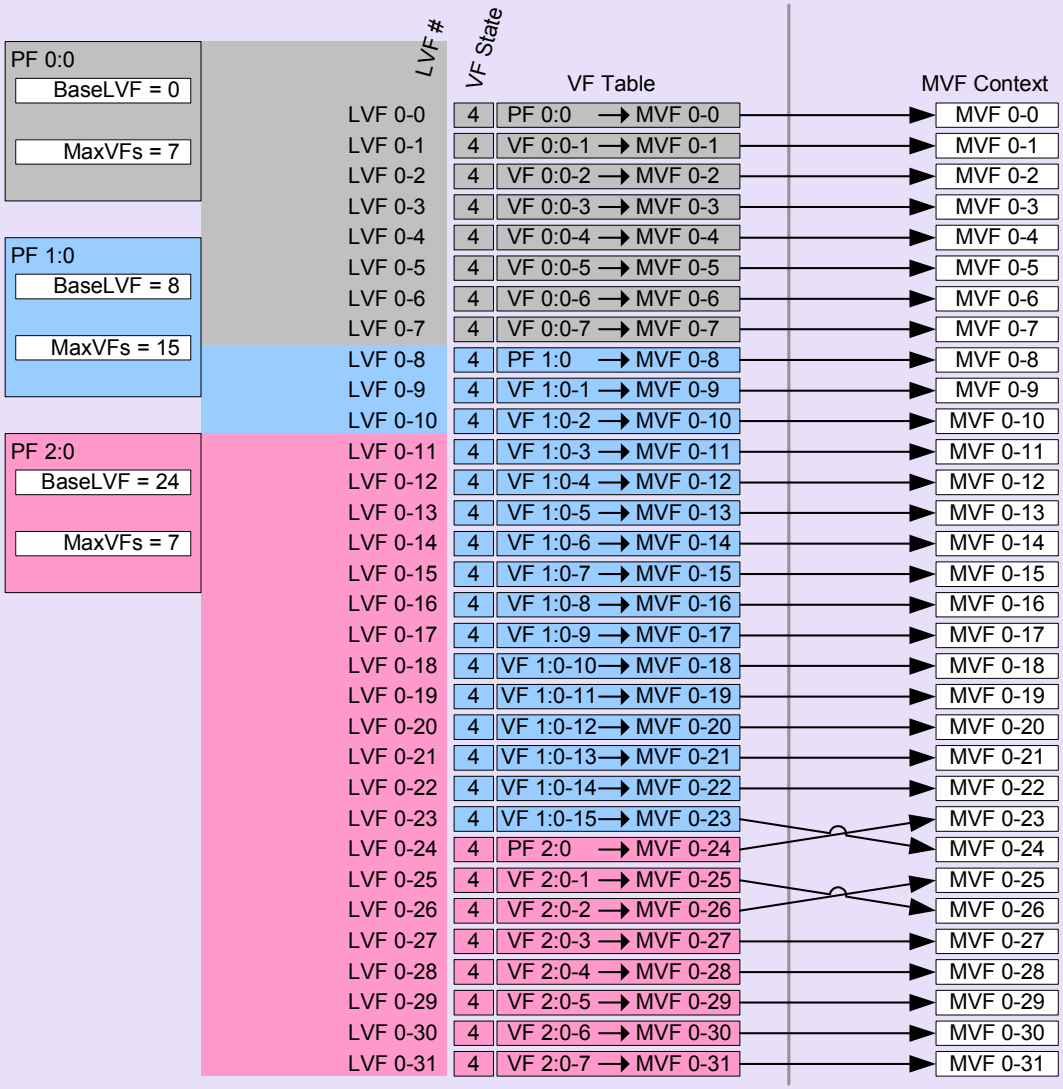
- Switch B: VS 2
  - ✓ Upstream: Port D PCIe (from RP 3)
  - ✓ Downstream 0: Port L VH 2
  - ✓ Downstream 1: Port M VH 2
  - ✓ Downstream 2: Port N PCIe
  - ✓ Downstream 3: Port Y VH 0 (to SW A)
  - ✓ Downstream 4: —
  - ✓ Downstream 5: —
- Switch A: VS 3
  - ✓ Upstream: Port X VH 0 (from SW B)
  - ✓ Downstream 0: Port G VH 3
  - ✓ Downstream 1: Port H VH 3
  - ✓ Downstream 2: Port I VH 3
  - ✓ Downstream 3: Port I VH 2

Configuring NumVH

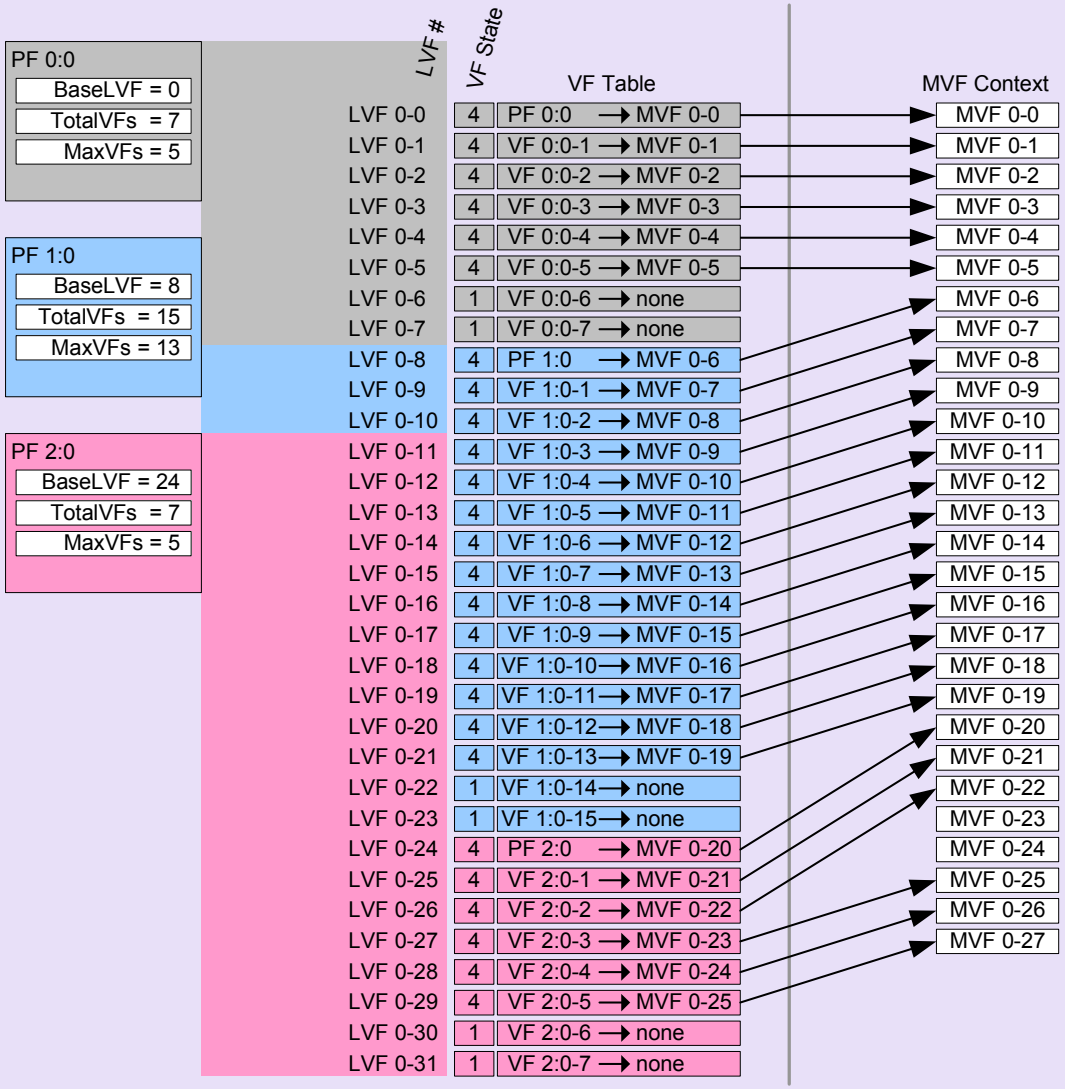
- Device G
  - ✓ MaxVHs = 8
  - ✓ NumVHs = 4
  - ✓ VH 0 → RP 0 VH 1 → **RP 2** VH 2 → **RP 1** VH 3 → RP 3
- Device H
  - ✓ MaxVHs = 8
  - ✓ NumVHs = 4
  - ✓ VH 0 → RP 0 VH 1 → **RP 1** VH 2 → **RP 2** VH 3 → RP 3
- Device I
  - ✓ MaxVHs = 8
  - ✓ NumVHs = **6**
  - ✓ VH 0 → RP 0 VH 1 → RP 2 VH 2 → **RP 3** VH 3 → **RP 3**
  - ✓ VH 4 → **None (Hot Plug)** VH 5 → **None (Hot Plug)**
- Device L
  - ✓ MaxVHs = 8
  - ✓ NumVHs = **3**
  - ✓ VH 0 → RP 2 VH 1 → RP 0 VH 2 → RP 3
- Device M
  - ✓ MaxVHs = 8
  - ✓ NumVHs = **3**
  - ✓ VH 0 → RP 2 VH 1 → RP 0 VH 2 → RP 3

- **NumVH values also programmed in Switch PF Table**

Configuring Device L: VF Mapping



Configuring Device M: VF Mapping and Migration



VH Enumeration

- MR-PCIM has finished configuring Switches and Devices
- MR-PCIM allows RP 1, RP 2 and RP 3 to start
  - ✓ Mechanism is Out-of-Scope
- Each RP performs traditional PCIe Enumeration
  - ✓ Establishes Bus Numbering, etc
- Each VS appears as PCIe Switch
- Each PF appears as PCIe Device (with optional SR-IOV Support)
  - ✓ Optional SR-PCIM runs *within each VH*

MR Config Space

Steve Glaser, NextIO Inc.

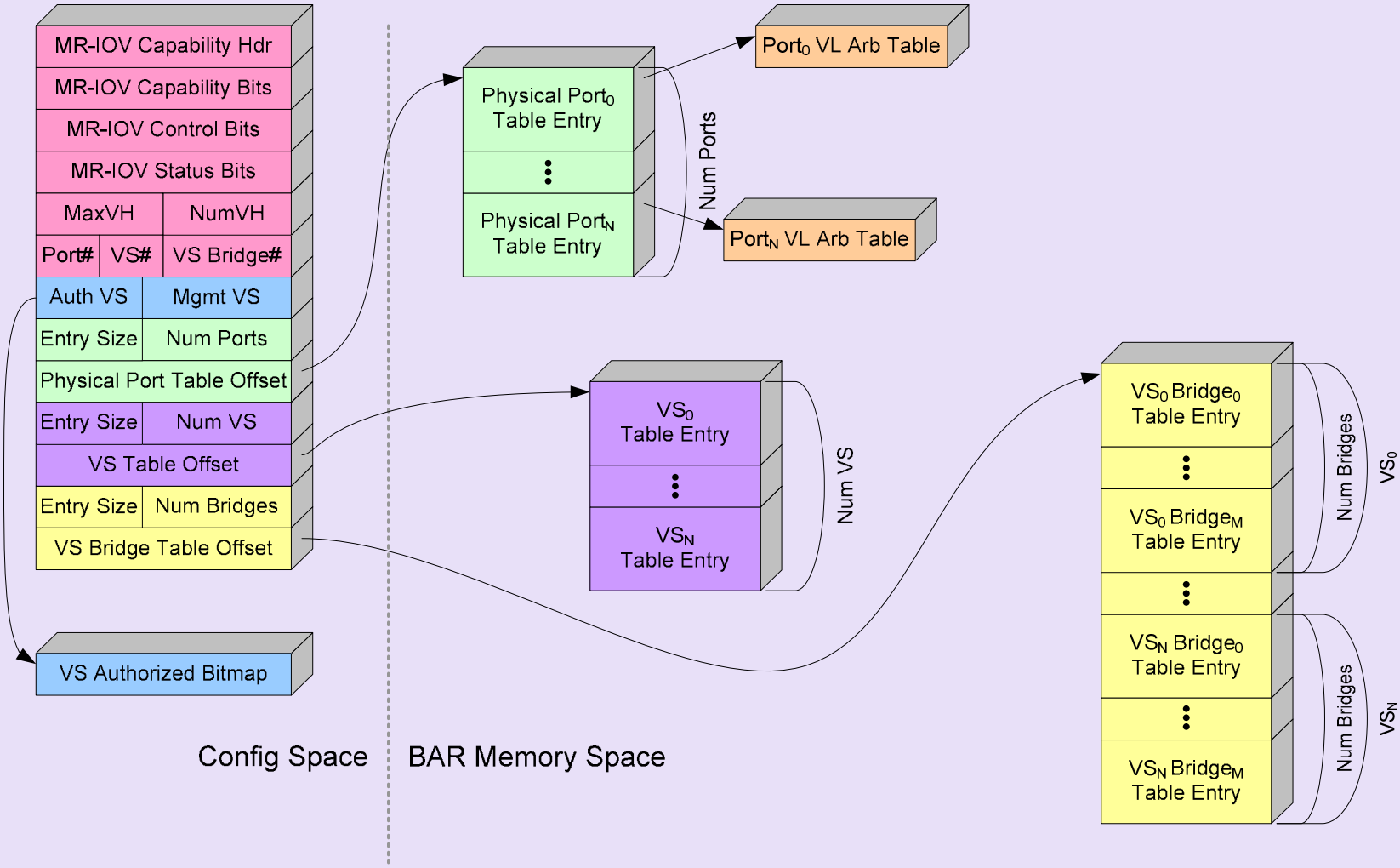
MR Switch Configuration Regs

- MR-IOV Capability block in Type 1 Config Headers
  - ✓ Each MR Switch has one or more
    - – Upstream ports of Authorized Virtual Switches
  - ✓ All MR-IOV Capabilities of switch describe same underlying hardware

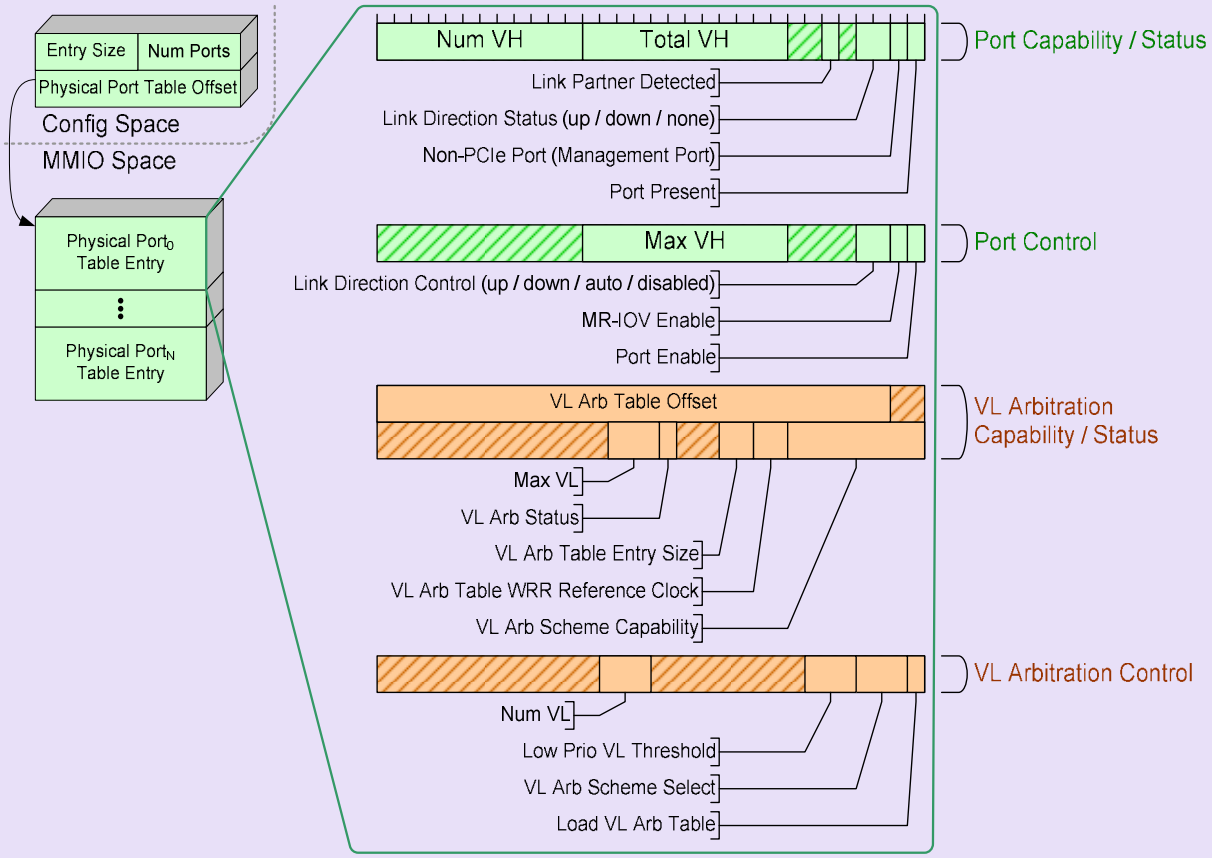
- Controls split into 4 groups:
  - ✓ Global MR Switch Controls
  - ✓ Per Port Controls
  - ✓ Per Virtual Switch Controls
  - ✓ Per Bridge on Virtual Switch Controls

  - ✓ Global MR Switch Controls in Config Space
  - ✓ Other Controls in Memory Space

Switch: MR-IOV Tables



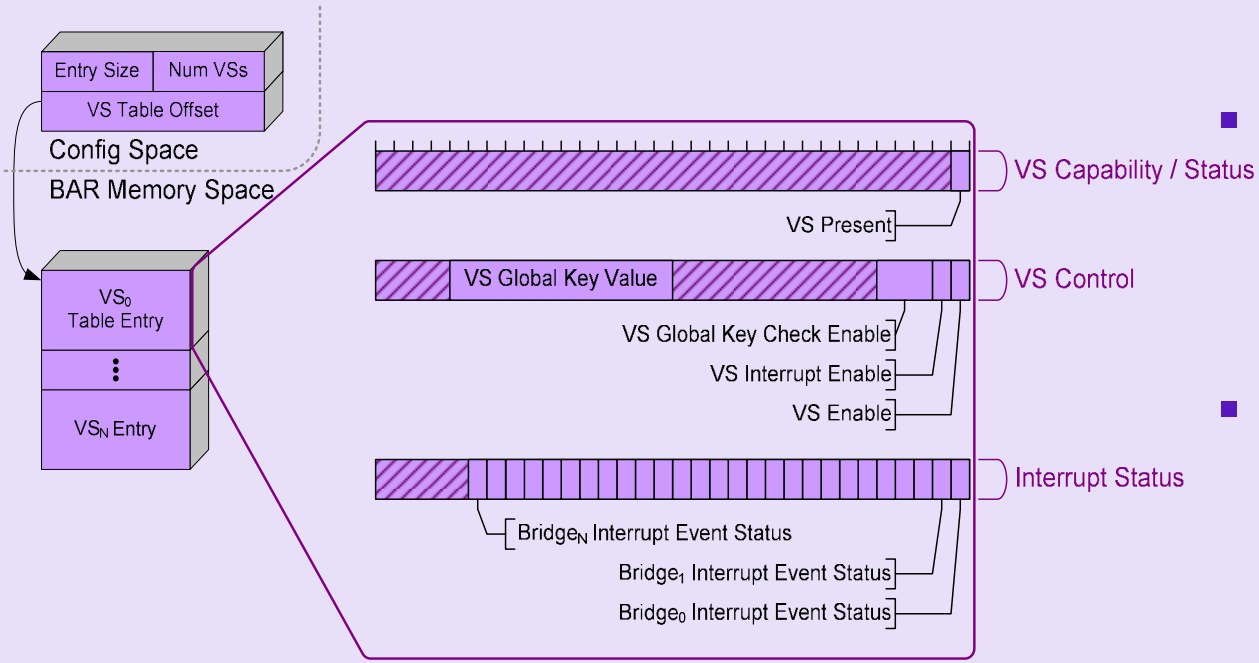
Switch: Physical Port Table



Three Knob Kinds:

- Link Mgmt
  - ✓ Link Direction / Status
- VH Mgmt
  - ✓ # VHs implemented
  - ✓ # VHs offered to MR upstream
  - ✓ # VHs enabled by MR upstream
- VL Mgmt
  - ✓ VL Arbitration Table (optional)
  - ✓ Like PCIe VC Arbitration

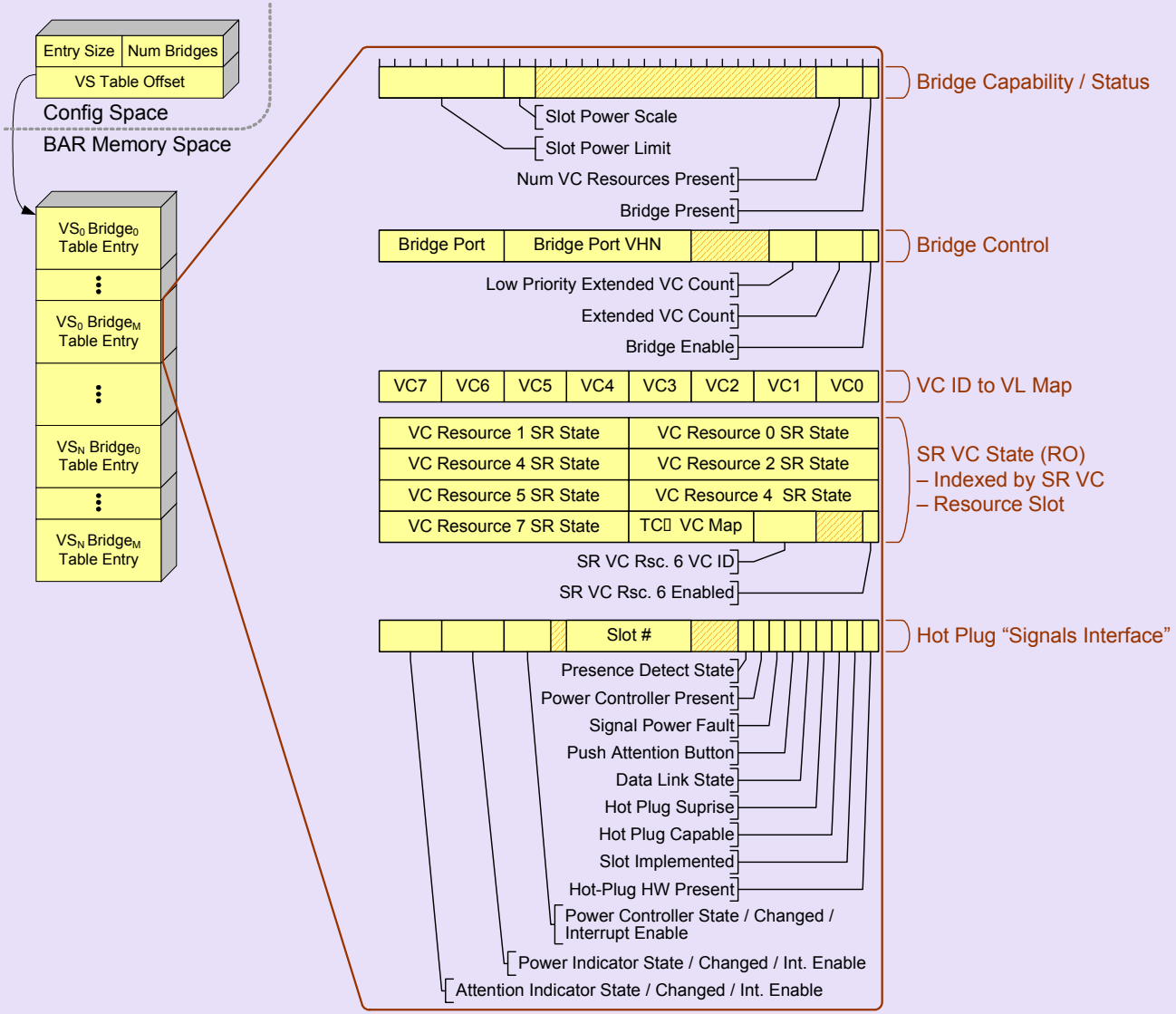
Switch: VS Table



Three knob kinds:

- Misc.
  - ✓ VS Present
  - ✓ VS Enable
- Global Key
  - ✓ Key Value
  - ✓ Check Enable bits
- Interrupts
  - ✓ Interrupt Enable
  - ✓ Per Bridge Interrupt Status
  - ✓ E.g. Hot Plug

Switch: VS Bridge Table



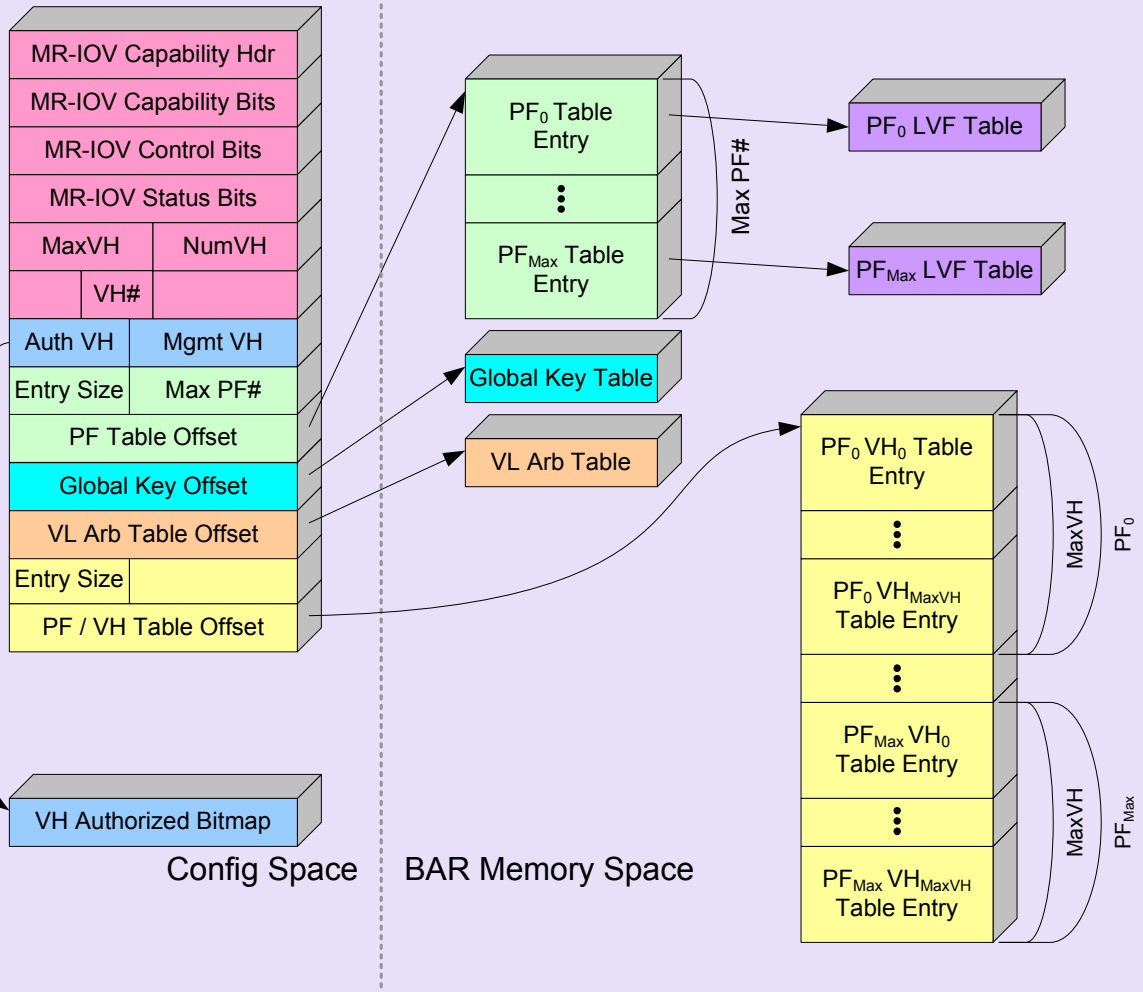
Four knob kinds:

- Bridge Mapping
  - ✓ Bridge Port & VHN
- VC Control
  - ✓ VC → VL Map
  - ✓ SR VC State
- Hot Plug Signals
  - ✓ Virtual "wires" from PCIe
- Misc.
  - ✓ Bridge Present
  - ✓ Bridge Enable
  - ✓ Slot Power, …

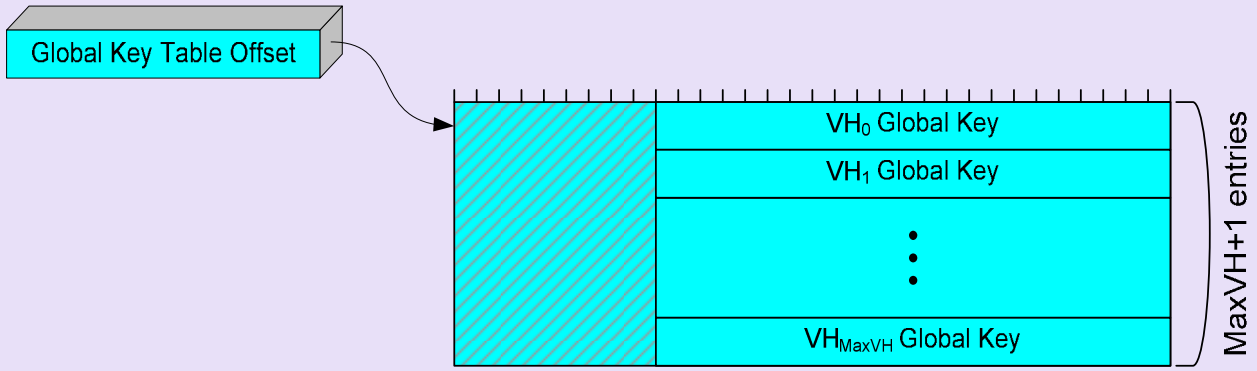
MR Device Configuration Regs

- MR-IOV Capability block in Type 0 Config Header
  - ✓ Visible in Function 0 after Reset
  - ✓ Visible in BF after MR-IOV Enabled
    - – Same underlying hardware
  - ✓ Visible in every Authorized VH
- Controls split into 5 groups:
  - ✓ Global MR Device Controls
  - ✓ Per VH Controls (Global Key)
  - ✓ Per PF Controls
  - ✓ Per PF / VH Controls
  - ✓ VL Arbitration Controls (optional)
  - ✓ Global MR Device Controls in Config Space
  - ✓ Other Controls in Memory Space

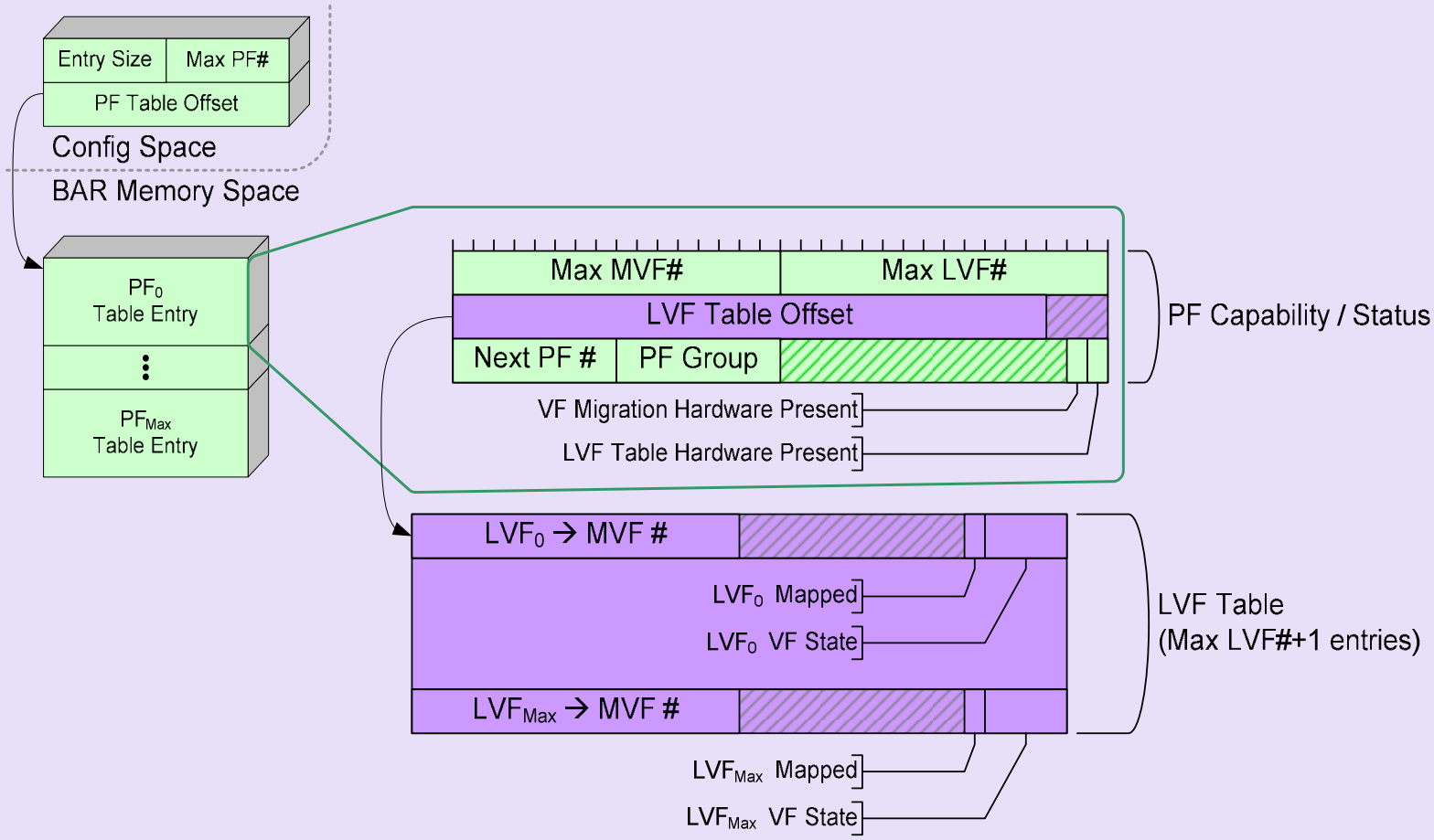
Device: MR-IOV Tables



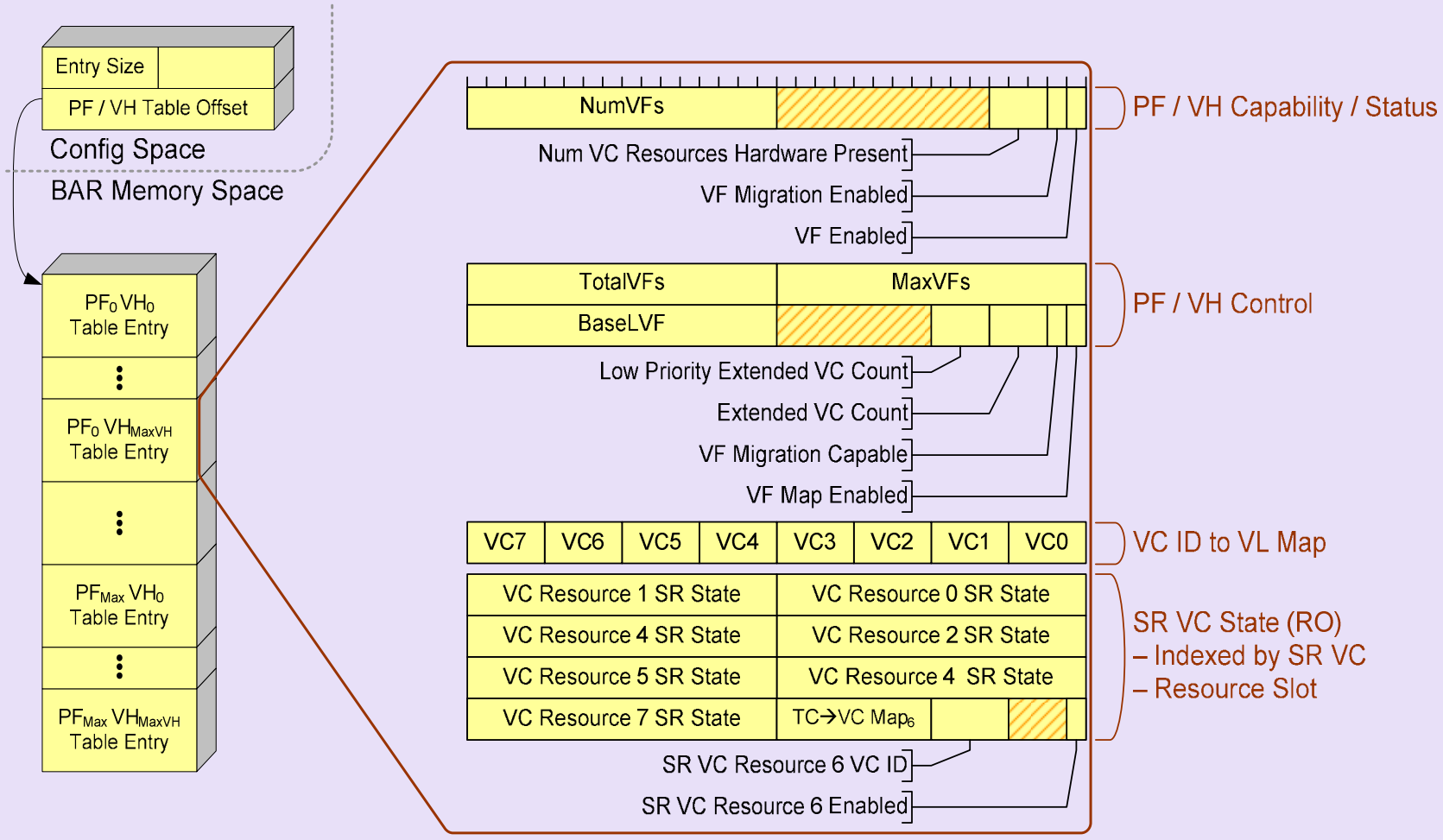
Device: Global Key Table



Device: PF Table
(and optional LVF Table)

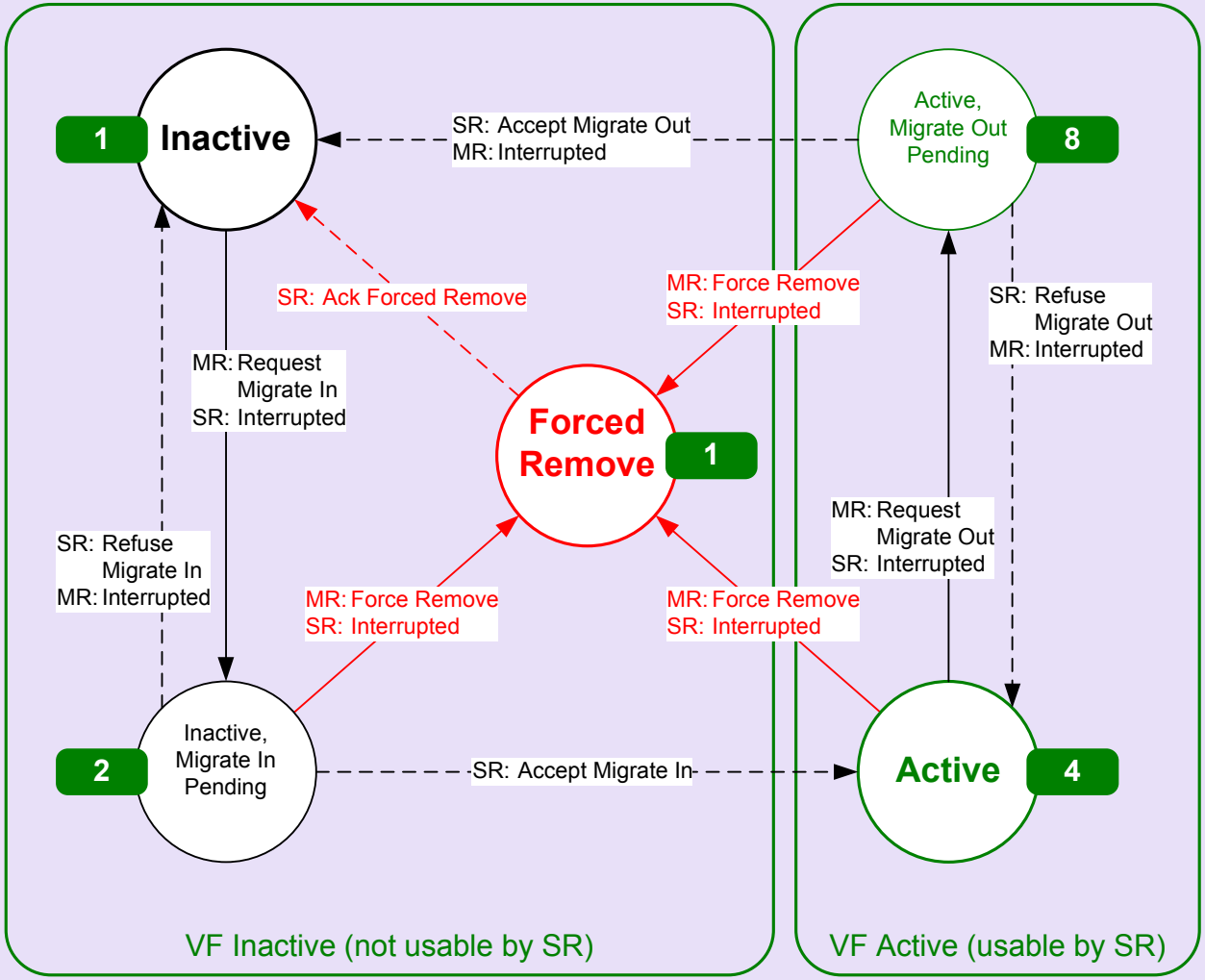


Device: PF / VH Table



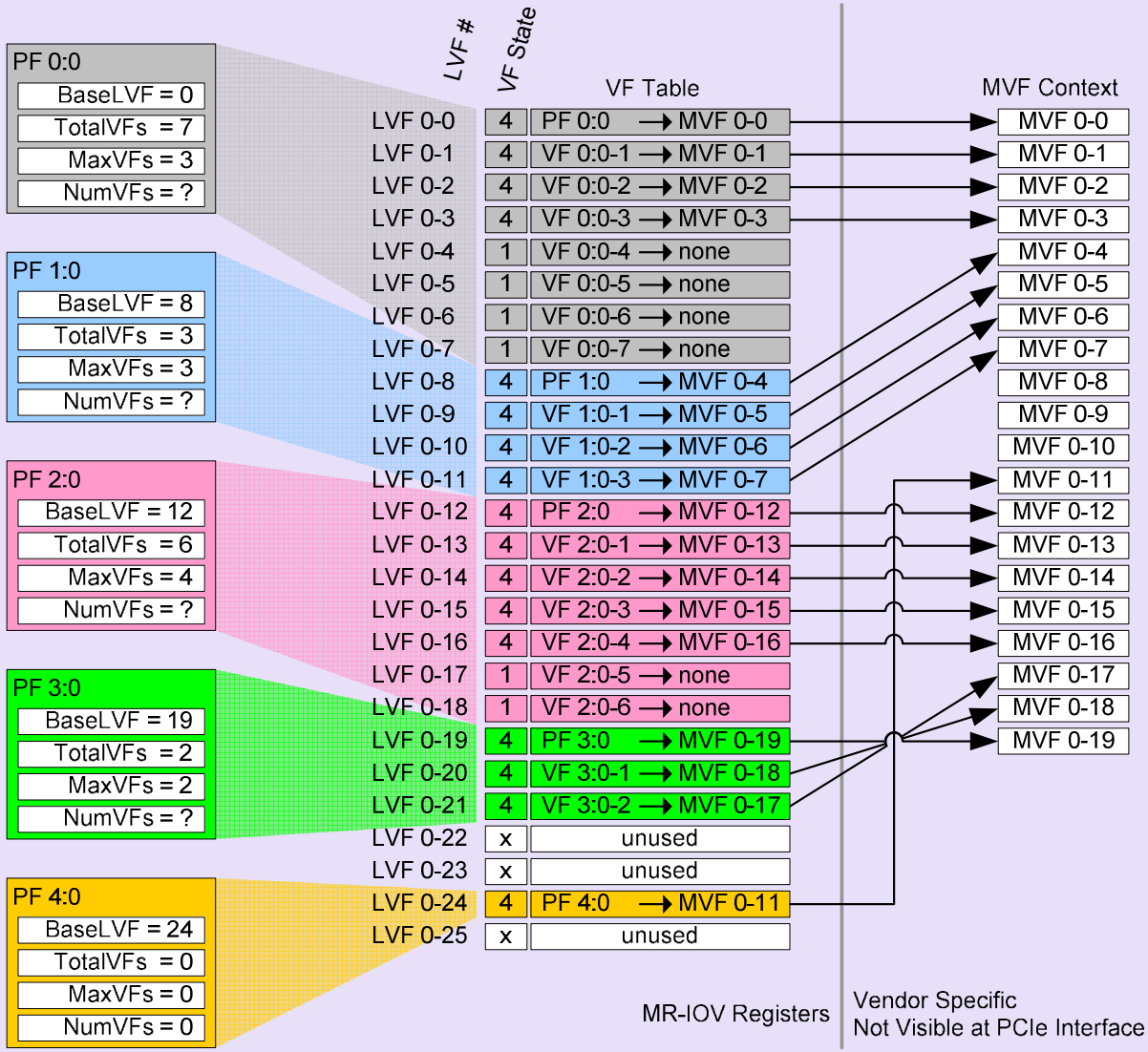
VF Migration

VF Migration State



- SR and MR PCIM are both part of migration
  - ✓ Both must enable
- MR Populates some VFs
  - ✓ Active state (== 4)
- MR leaves some VFs unpopulated
  - ✓ Inactive state (== 1)
- Migrate In
  - ✓ Populate then 1 → 2 → 4
- Migrate Out
  - ✓ 4 → 8 → 1 then depopulate
- Forced Remove
  - ✓ 2 / 4 / 8 → 1
  - ✓ Ungraceful – like surprise data link down

VF Mapping: LVF Table



Questions


PCI
SIG®