Tag-less Virtual Ethernet Port Aggregator (VEPA) Proposal

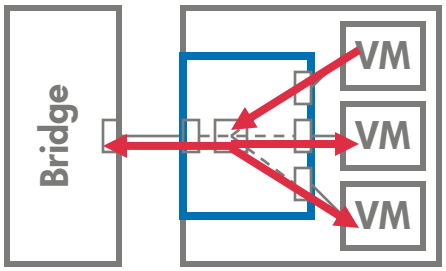
January 2009

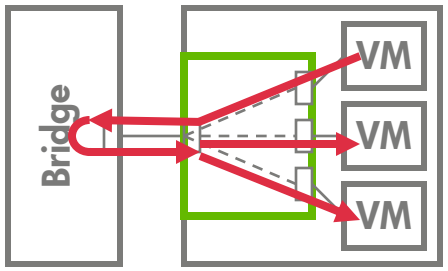
Chuck Hudson (HP) c.hudson@hp.com
Paul Congdon (HP) ptcongdon@ucdavis.edu

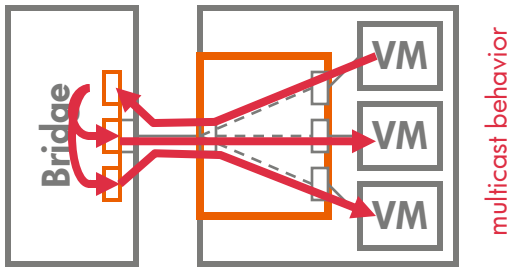
Motivation

- Enable robust bridge features to individual virtual machines
  - Network controls / ACLs
  - Network monitoring & security
  - Private VLANs
- Coordinated management of the network edge
  - Physical servers
  - Virtual servers
- Simplify data center management
- Rapid industry adoption

Summary of Possible Technical Approaches







Virtual Ethernet Bridge (VEB)

uses MAC+VID to steer frames

- Emulates 802.1 Bridge
- Limited controls
- Managed by station
- Works with all existing bridges
- No changes to existing frame format.
- Open-ended changes to NIC

Tag-less VEPA

uses MAC+VID to steer frames

- Extends 802.1 Bridge
- Advanced controls
- Managed by bridge
- Works with many existing bridges
- No changes to existing frame format.
- Limits NIC changes

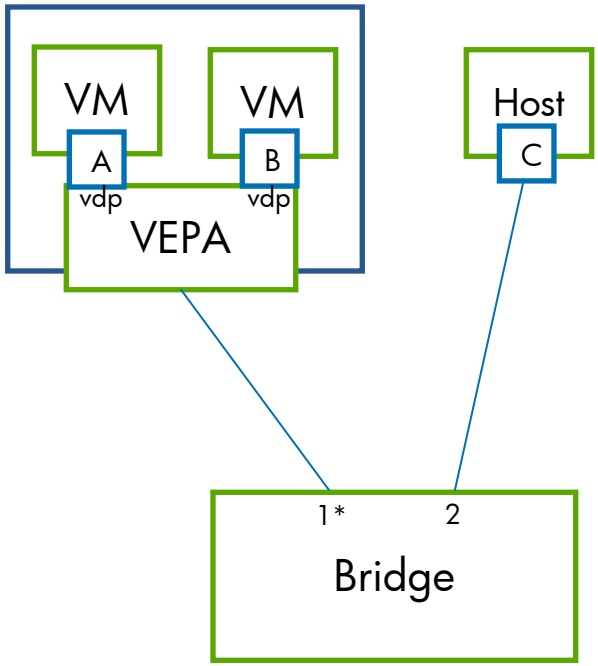
Tagged

uses new tag to steer frames

- Extends 802.1 Bridge
- Advanced controls
- Managed by bridge
- Works with few or no existing bridges
- Changes to existing frame format.
- Limits NIC changes

Tag-less 101

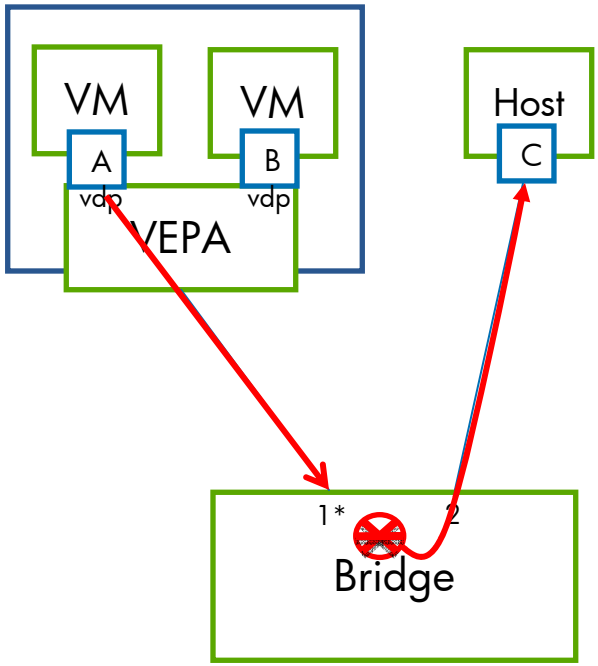
Virtual Ethernet Port Aggregator (tagless)



1. Provides Multiple VEPA Device Ports (vdp) as vNICS to Virtual Machines
2. Each VDP is configured as individual NIC (i.e. MAC addr, Multicast addrs, VLAN tags, or passthru). VEPA aggregates configurations.
3. May support all traditional NIC features (e.g. TCP Checksum, RSS, Large Segment Send)
4. Does NOT perform Local Bridging. Not a Virtual Ethernet Bridge (VEB)
5. Sends all outbound traffic to the wire
6. Replicates received mcast/bcast traffic
7. VLAN aware
8. May provide QoS and BW management
9. Invoked by special Bridge mode negotiation

Note: This proposal does NOT require new tags, but could work with them.

VEPA Forwarding



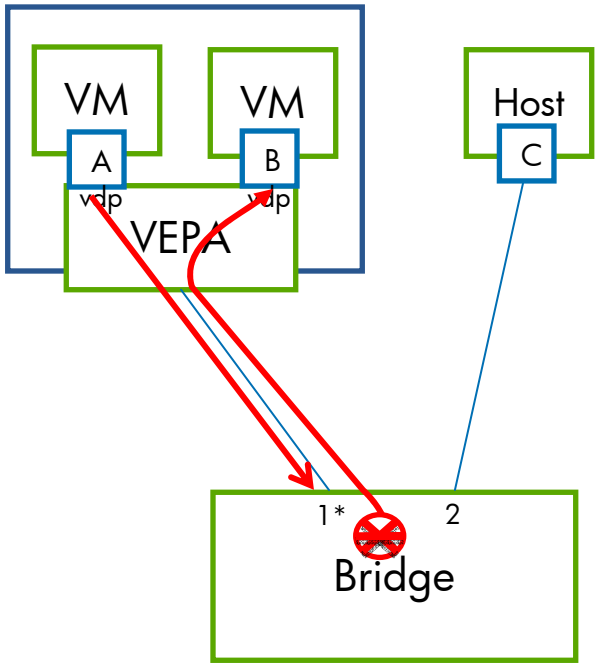
1. A->C
2. A->B
3. A->Bcast
4. C->Bcast

Bridge Address Table

Address	Port
A	1
B	1
C	2

* = Bridge Port Configured for VEPA attach

VEPA Forwarding



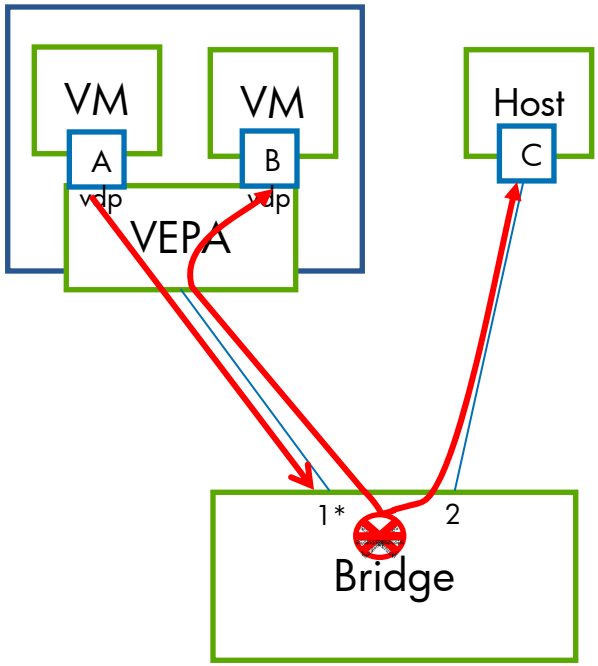
1. A->C
2. A->B
3. A->Bcast
4. C->Bcast

Bridge Address Table

Address	Port
A	1
B	1
C	2

* = Bridge Port Configured for VEPA attach

VEPA Forwarding



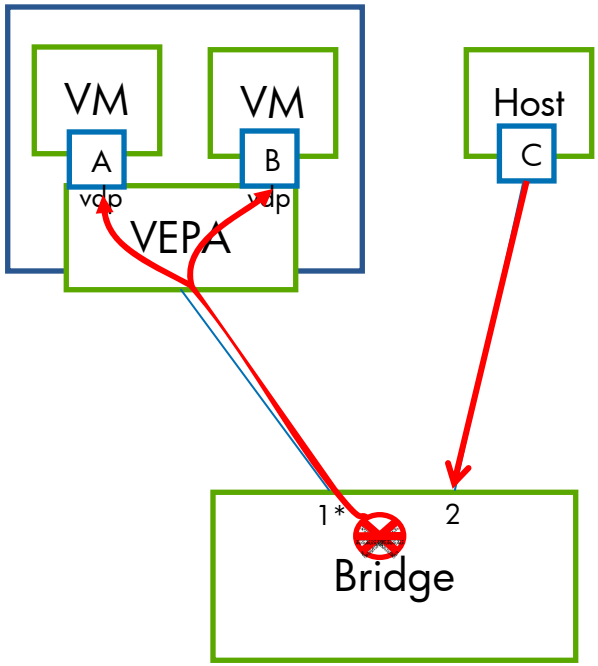
1. A->C
2. A->B
3. A->Bcast
4. C->Bcast

Bridge Address Table

Address	Port
A	1
B	1
C	2

* = Bridge Port Configured for VEPA attach

VEPA Forwarding



1. A->C
2. A->B
3. A->Bcast
4. C->Bcast

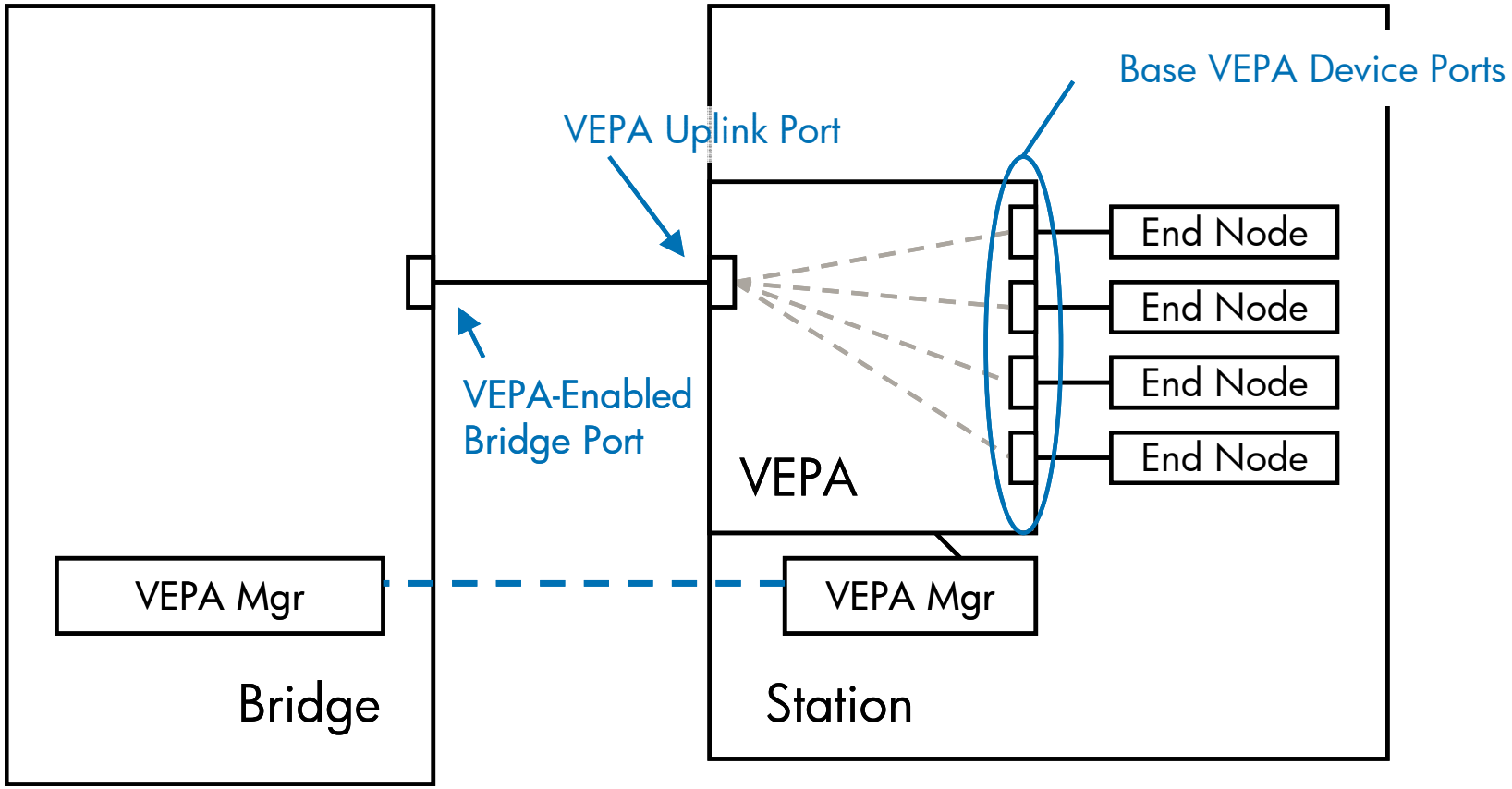
Bridge Address Table

Address	Port
A	1
B	1
C	2

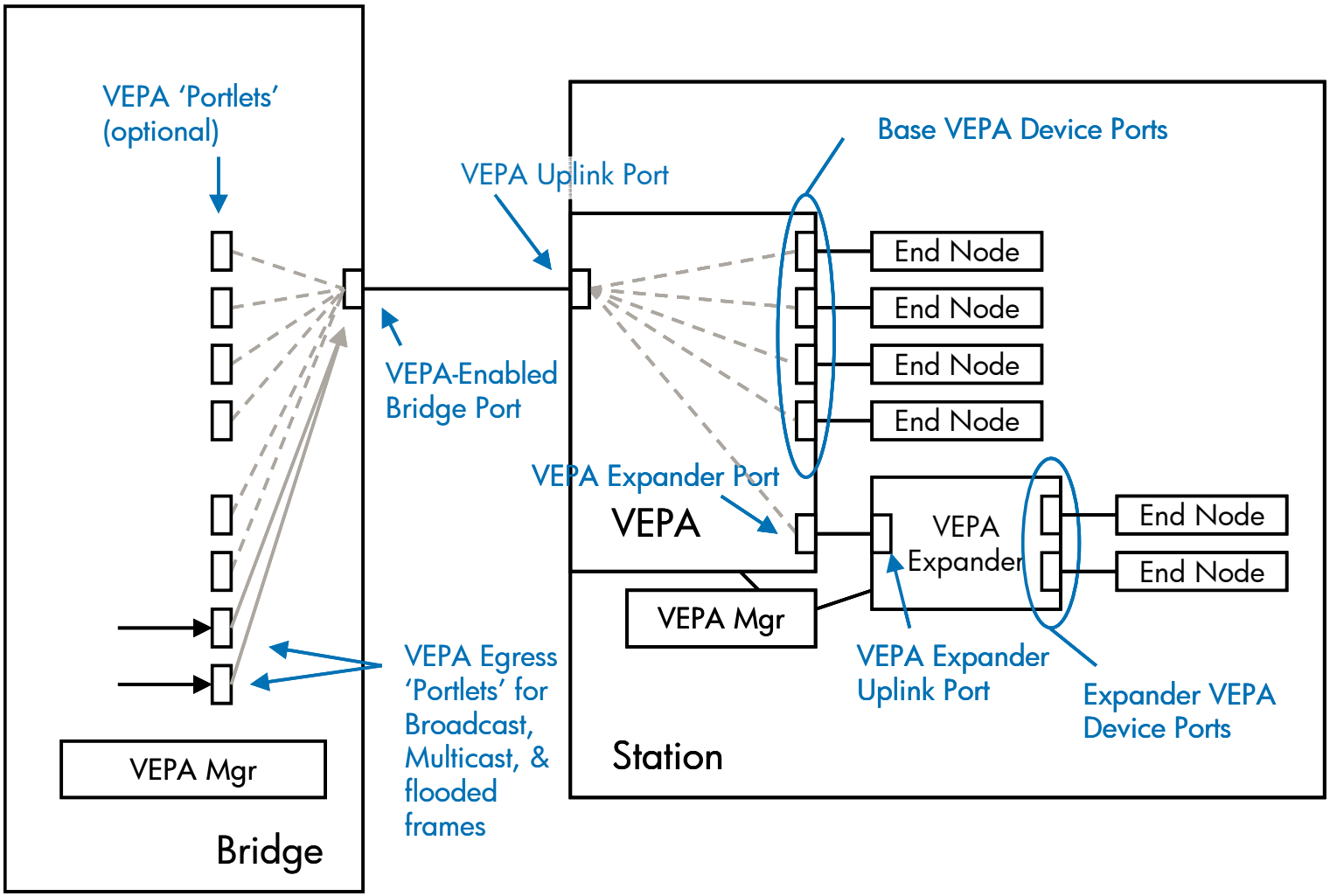
* = Bridge Port Configured for VEPA attach

VEPA Elements

Key VEPA Terms



Additional VEPA Terms



Basic Tag-less VEPA Construction

- Each VEPA has
  - A single, active VEPA Uplink Port
  - 1 to n VEPA Device Ports
  - 0, 1, or more VEPA Expander Ports
  - Station VEPA Manager & VEPA Address Table
- Connected to VEPA-enabled Bridge Port
  - VEPA 'Portlets' (optional)
  - Egress 'Portlets' (optional)
  - Bridge VEPA Manager
- A station may have multiple VEPAs

VEPA Device Ports

- Each VEPA Device Port
  - May be implemented as a PCI virtual function
  - Has one or more statically-identified MAC addresses
  - Movement of MAC addresses coordinated through VEPA Managers
- VEPA Device Ports are 'NIC Configuration Aware'
  - Of MAC addresses
  - Of MAC listening entries (multi-cast and unicast)
- Configured via Station VEPA Manager
  - 1 or more specific MAC addresses (by station)
  - VLAN tagging behavior*
  - Priority tagging behavior*
- Forwards incoming frames to VEPA uplink
  - May set VLAN/Priority based on settings
- Receives frames from VEPA uplink
  - May remove VLAN/Priority tag based on settings

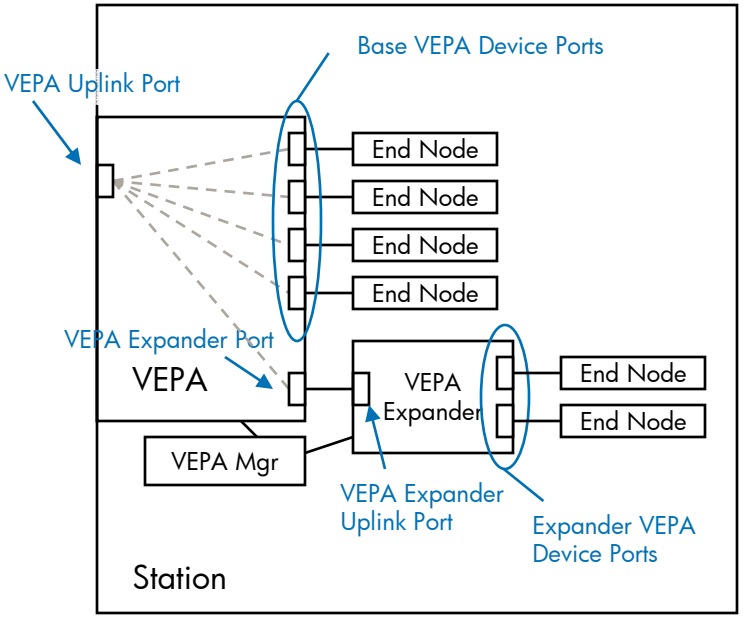
* Can be set by station or bridge

VEPA Uplink Port

- Single VEPA Uplink per VEPA
  - May be LAG
  - Has a MAC address (for capability exchange)
  - May implement ETS queues
- Settings
  - VEPA MAC address
  - Acceptable frame types
    - Only VLAN tagged
    - Untagged, Pri tagged
    - All frames
  - PVID
  - Egress VLAN IDs (aggregate of the VDP VIDs)

VEPA Expander

- Usually software (operating mode of vswitch)
- Extends beyond limits of HW VEPA
  - # of VEPA Device Ports
  - # of VEPA Address Table Entries
- Consists of
  - One VEPA Expander Uplink Port
  - One to m Expansion VEPA Device Ports
  - Expander VEPA Address Table
- Forwards frames from VDPs to VEPA Uplink
- Sends (replicating as necessary) from Expander Uplink Port to expansion VDPs
- Linked to Station VEPA Manager
  - Configuration of VEPA Device Ports
  - Contribute to VEPA Capability Exchange



VEPA-enabled Bridge Port…

- The port is enabled for 'turn-around' forwarding of
  - Multicast
  - Broadcast
  - Flooded Unicast
- Unicast destinations per forwarding table
- The bridge may implement controls and features via
  - VEPA Device Port configuration
    (VLAN ID, Private VLANs, Priority Settings, MAC filtering)
  - Portlets (ACLs, Statistics)
  - Address table entries (IGMP snooping)

VEPA 'Portlets'

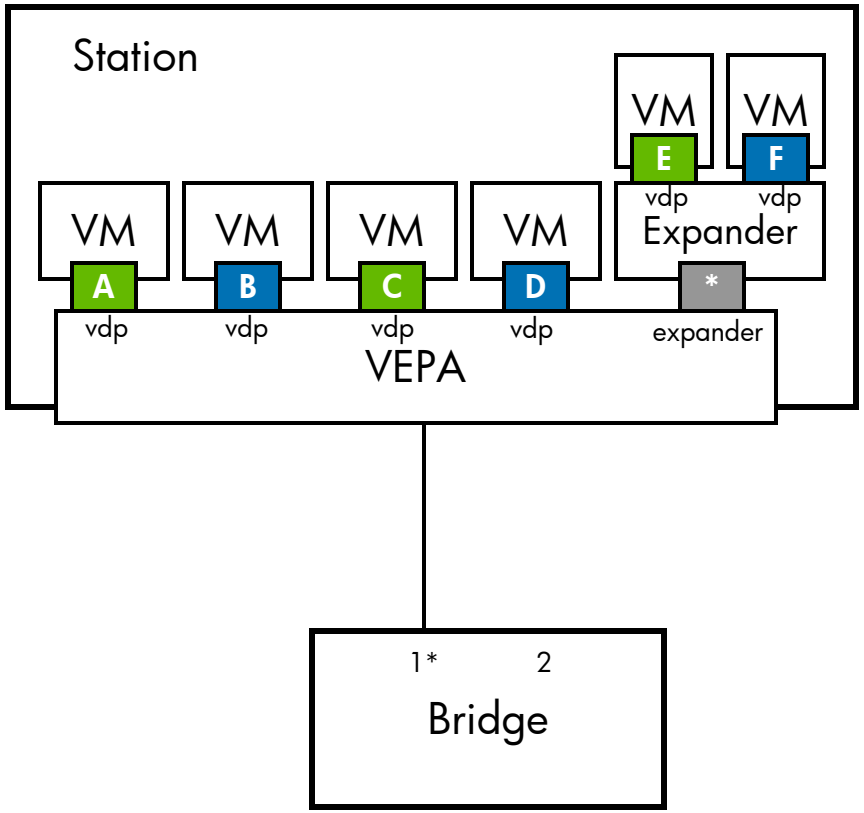
- Optional (can be simulated by rules engines)
- Useful in simplifying ACLs & statistics collection
- VEPA Portlets
  - Associated with VDP MAC address(s)
  - Identifies incoming frames by SRC MAC
  - Identifies outgoing unicast frames by DST MAC
- Egress Portlets
  - Extra controls & statistics on broadcast, multicast, and flooded frames

VEPA Address Table Management

Address Table Management

- Managed by Station VEPA Manager
  - Information coordinated with bridge via VEPA Capability Exchange
- Static settings (no learning)
- Driven by NICs
  - VM NIC driver register for unicast/multicast listens
    - Fully-supports Locally-Assigned MAC Addresses (LAA)
  - Station VEPA manager receives request
  - Station VEPA manager creates/updates table entries
- Multicast entries may be driven by Bridge (IGMP snooping)
  - Bridge intercepts join/leave messages
  - Creates/updates/deletes address table multicast entry

VEPA Address Table Example



* = Bridge Port Configured for VEPA attach

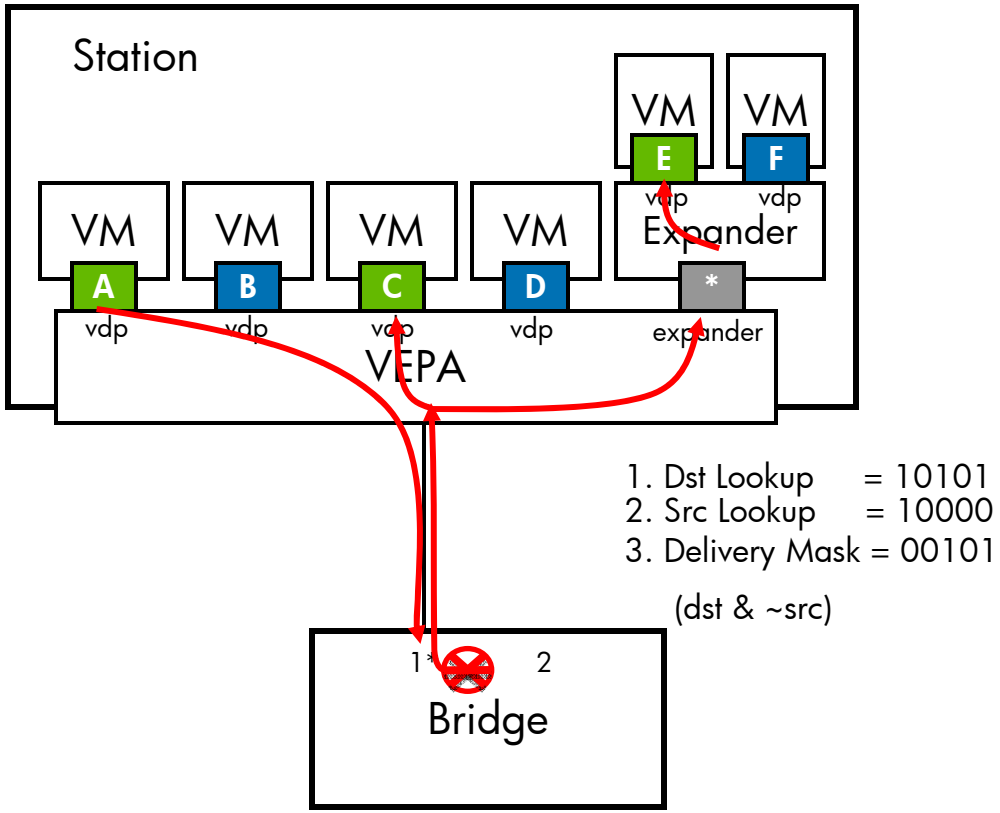
Example: Base VEPA Address Table

Address	VLAN	Mask (ABCD *)
A	1	1000 0
B	2	0100 0
C	1	0010 0
D	2	0001 0
Bcast	1	1010 1
Bcast	2	0101 1
Mcast1	1	1010 1
Mcast1	2	0100 1
Mcast2	2	0101 1
Unk Mcast	1	0000 1
Unk Mcast	2	0000 1
Unk Ucast	1	0000 1
Unk Ucast	2	0000 1

VLAN 1 Tag Mask = UUUUT
VLAN 2 Tag Mask = UUUUT

VEPA Address Table Example

A -> Bcast



Note: Bridge should echo IGMP packets too

Example: Base VEPA Address Table

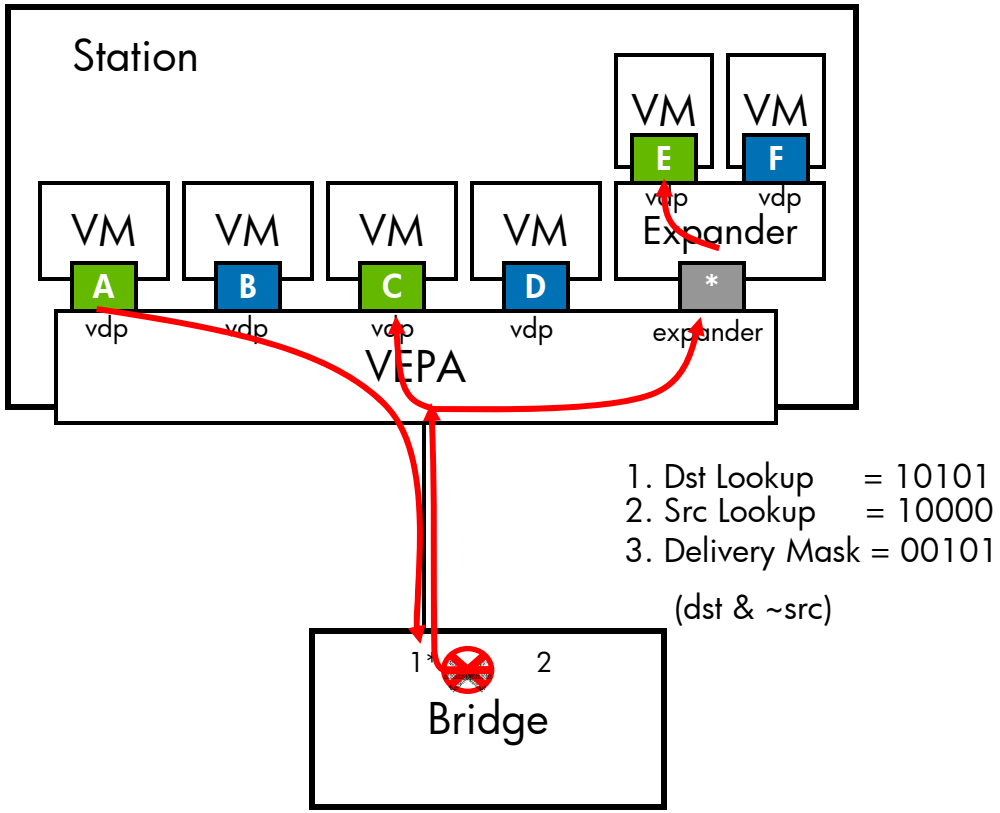
Address	VLAN	Mask (ABCD *)
A	1	1000 0
B	2	0100 0
C	1	0010 0
D	2	0001 0
Bcast	1	1010 1
Bcast	2	0101 1
Mcast1	1	1010 1
Mcast1	2	0100 1
Mcast2	2	0101 1
Unk Mcast	1	0000 1
Unk Mcast	2	0000 1
Unk Ucast	1	0000 1
Unk Ucast	2	0000 1

* = Bridge Port Configured for VEPA attach

VLAN 1 Tag Mask = UUUUT
VLAN 2 Tag Mask = UUUUT

VEPA Address Table: Multicast Entries

A -> Mcast1



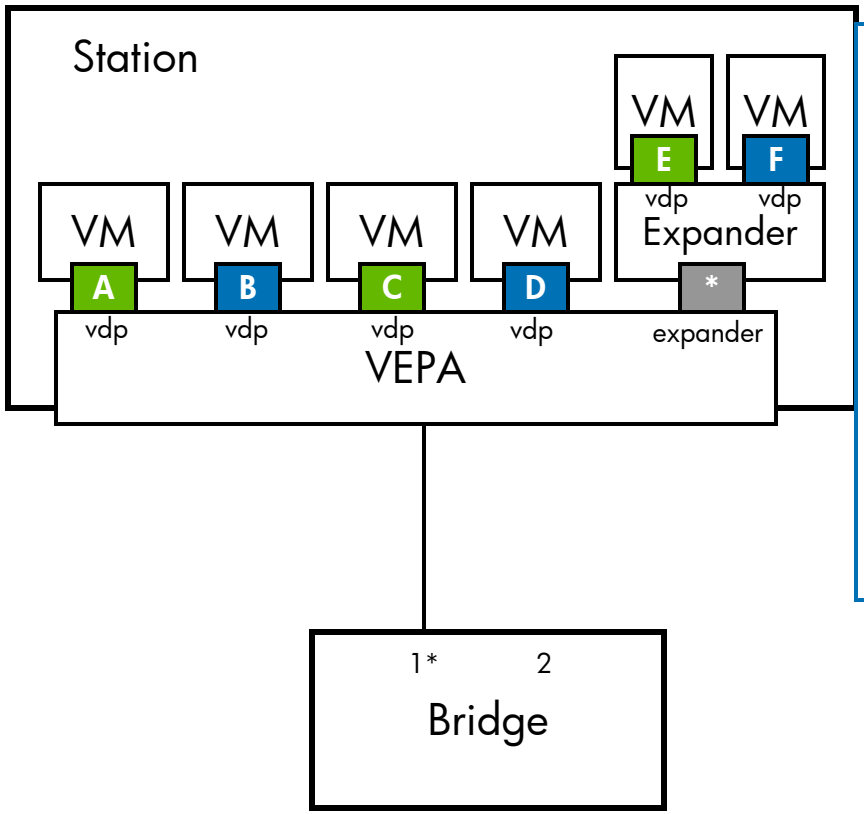
Example: Base VEPA Address Table

Address	VLAN	Mask (ABCD *)
A	1	1000 0
B	2	0100 0
C	1	0010 0
D	2	0001 0
Bcast	1	1010 1
Bcast	2	0101 1
Mcast1	1	1010 1
Mcast1	2	0100 1
Mcast2	2	0101 1
Unk Mcast	1	0000 1
Unk Mcast	2	0000 1
Unk Ucast	1	0000 1
Unk Ucast	2	0000 1

* = Bridge Port Configured for VEPA attach

VLAN 1 Tag Mask = UUUUT
VLAN 2 Tag Mask = UUUUT

VEPA Address Table: Unknown addresses



Example: Base VEPA Address Table

Address	VLAN	Mask (BCD *)
		000 0
		00 0
		010 0
		001 0
		010 1
		01 1
		010 1
		00 1
Mcast2	2	0101 1
Unk Mcast	1	0000 1
Unk Mcast	2	0000 1
Unk Ucast	1	0000 1
Unk Ucast	2	0000 1

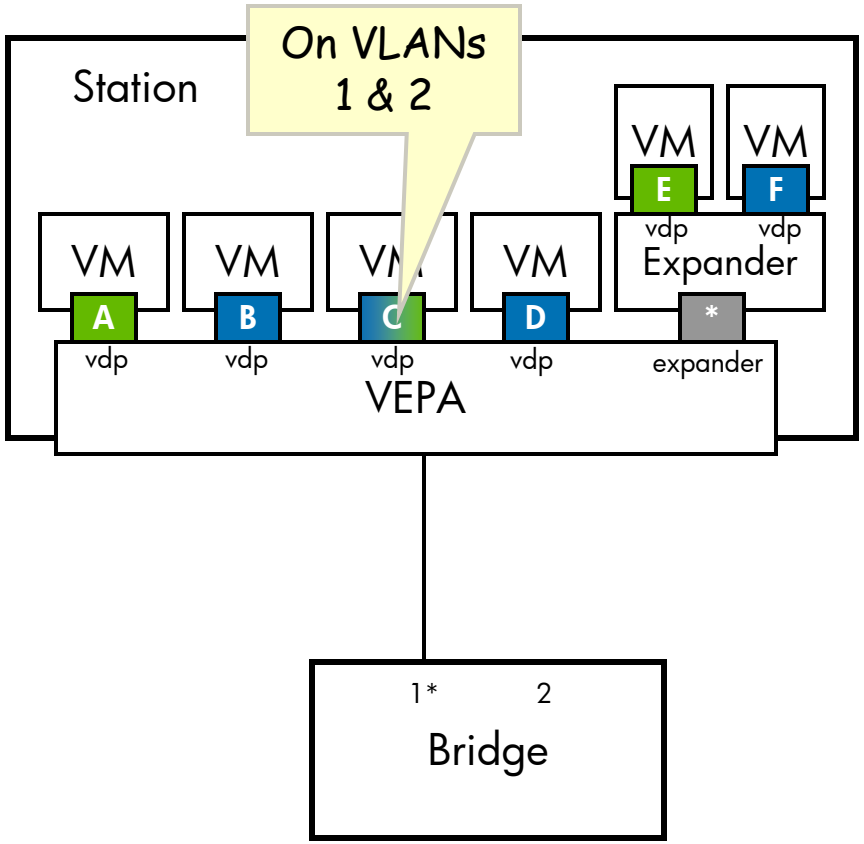
Unknown Multicast entries allow for multicast handling when there are excessive entries, promiscuous multicast listens, and steering of multicast entries to expander port.

Unknown unicast entries needed to steer packets to expander port(s). Also allows for support of promiscuous listen or monitoring ports.

* = Bridge Port Configured for VEPA attach

VLAN 1 Tag Mask = UUUUT
VLAN 2 Tag Mask = UUUUT

Multiple VLANs on VDP



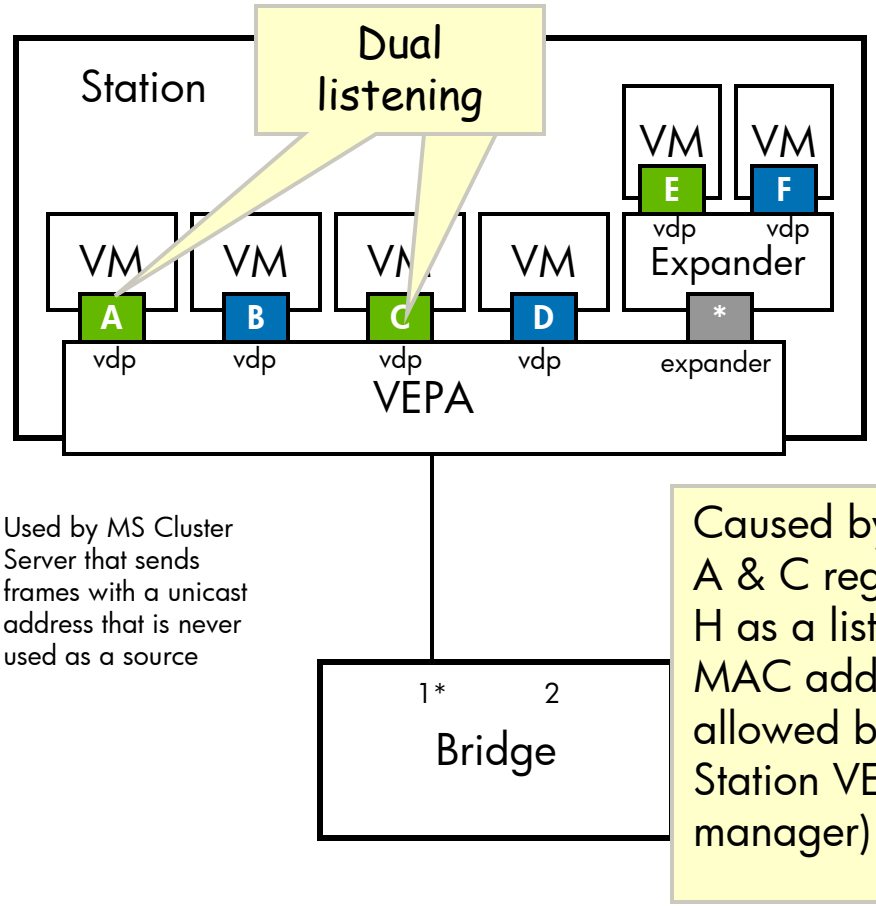
* = Bridge Port Configured for VEPA attach

Example: Base VEPA Address Table

Address	VLAN	Mask (ABCD *)
A	1	1000 0
B	2	0100 0
C	1	0010 0
C	2	0010 0
D	2	0001 0
Bcast	1	1010 1
Bcast	2	0101 1
Mcast1	1	1010 1
Mcast1	2	0100 1
Mcast2	2	0101 1
Unk Mcast	1	0000 1
Unk Mcast	2	0000 1
Unk Ucast	1	0000 1
Unk Ucast	2	0000 1

VLAN 1 Tag Mask = UUTUT
VLAN 2 Tag Mask = UUTUT

VDPs in Dual Listening Mode



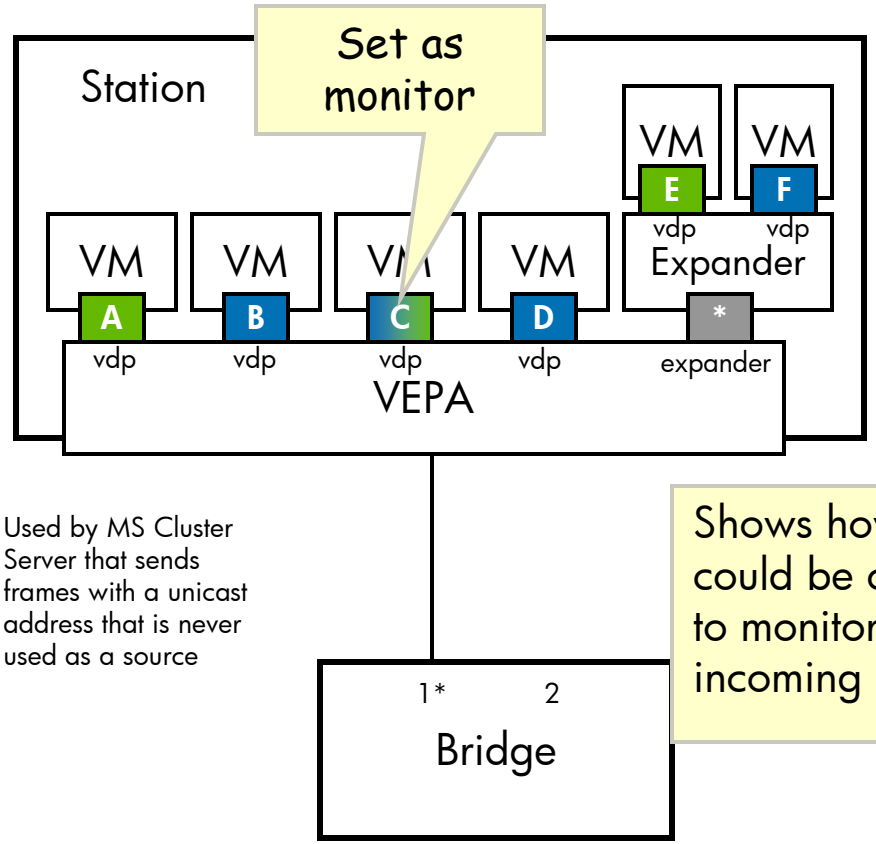
Example: Base VEPA Address Table

Address	VLAN	Mask (ABCD *)
A	1	1000 0
B	2	0100 0
C	1	0010 0
D	2	0001 0
H	1	1010 0
Bcast	1	1010 1
Bcast	2	0101 1
Mcast1	1	1010 1
Mcast1	2	0100 1
Mcast2	2	0101 1
Unk Mcast	1	0000 1
Unk Mcast	2	0000 1
Unk Ucast	1	0000 1
Unk Ucast	2	0000 1

* = Bridge Port Configured for VEPA attach

VLAN 1 Tag Mask = UUUUT
VLAN 2 Tag Mask = UUUUT

VDP in Monitor Mode



Used by MS Cluster Server that sends frames with a unicast address that is never used as a source

Shows how a VDP could be configured to monitor any or all incoming frames.

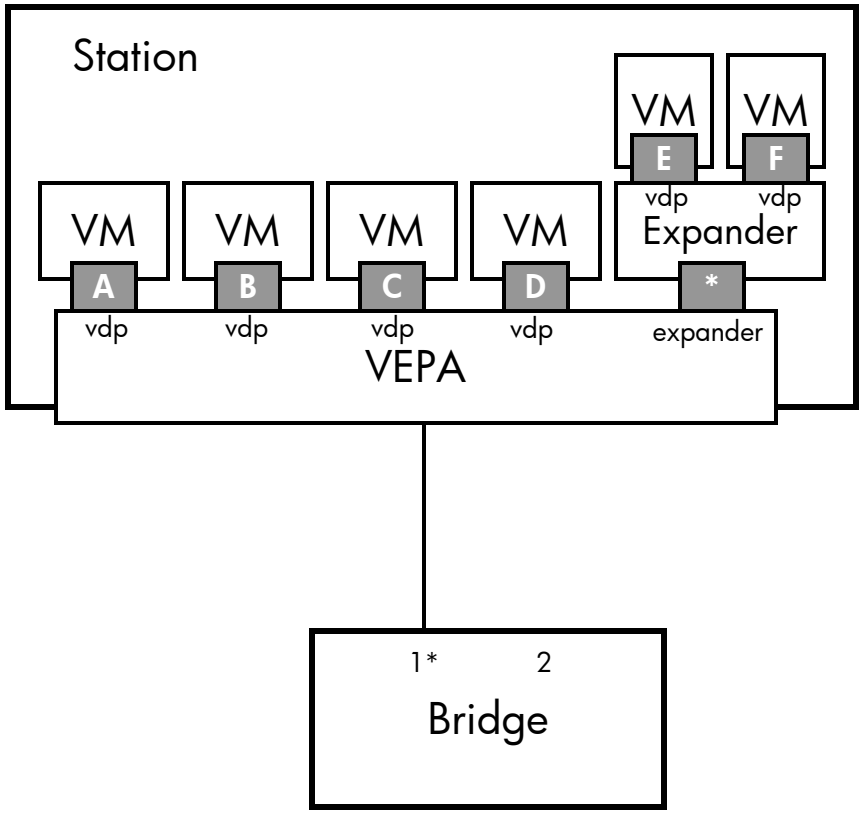
Example: Base VEPA Address Table

Address	VLAN	Mask (ABCD *)
A	1	1010 0
B	2	0110 0
C	1	0010 0
D	2	0011 0
Bcast	1	1010 1
Bcast	2	0111 1
Mcast1	1	1010 1
Mcast1	2	0110 1
Mcast2	2	0111 1
Unk Mcast	1	0010 1
Unk Mcast	2	0010 1
Unk Ucast	1	0010 1
Unk Ucast	2	0010 1

* = Bridge Port Configured for VEPA attach

VLAN 1 Tag Mask = UUTUT
VLAN 2 Tag Mask = UUTUT

VEPA Default Configuration (no VLAN tags)



* = Bridge Port Configured for VEPA attach

Example: Base VEPA Address Table

Address	VLAN	Mask (ABCD *)
A	1	1000 0
B	1	0100 0
C	1	0010 0
D	1	0001 0
Bcast	1	1010 1
Mcast1	1	0100 1
Mcast2	1	0101 1
Unk Mcast	1	0000 1
Unk Ucast	1	0000 1

Uplink configured as untagged

No VLAN or priority tagging

VLAN 1 Tag Mask = UUUUU

Configuration

VEPA Capability Exchange

- Between Station VEPA Manager and Bridge VEPA Manager
- Exchange VEPA capabilities, configuration
- Re-occurs as needed to keep bridge station up to date
  - Add, move, change of End Nodes
- Initial sequence
  - Establish link
  - Authenticate the link
    - Based on the VEPA Uplink's MAC address
    - Should allow for: MAC Auth, 802.1x, MACSEC
  - Link Aggregation Control Protocol (LACP) as appropriate
  - VEPA Capability Exchange

VEPA Capability Exchange

- Station → Bridge
  - VEPA Capabilities
    - Mode: Request/require: Tag-less, VEPA Tagged
    - # of base device ports
    - # of VEPA table entries
    - Level(s) of control
  - VEPA General Settings
    - Bridge vs. Station Control of VLAN ID
    - Bridge vs. Station Control of pri
  - Device Ports (Port Listing)
    - Port Number
    - Port Type (Base, Expander)
    - MAC addresses (as assigned by Station)
    - Settings
      - Acceptable Frame Types
      - PVID
      - VLAN IDs**
      - Ingress VID Filtering
      - Priority Settings
  - Address Table Entries (Typically Multicast)
    - Address, VLAN ID, Receiver Ports/Mask
  - Updates

- Station ← Bridge
  - VEPA Capabilities
    - Mode: Request/require: Tag-less, Tagged
    - Total # of supported device ports
    - Total # of supported address entries
  - VEPA General Settings
    - Echo: Control of VLAN ID
    - Echo: Control of priority
  - Device Port Setting (Port Listing)
    - Port Number
    - Echo/control settings
      - Acceptable Frame Types
      - PVID
      - VLAN IDs**
      - Ingress VID Filtering
      - Priority Settings
  - Address Table Entries (Typically Multicast)
    - Address, VLAN ID, Receive Ports/Mask
    - ….
  - Updates

** Could be done with VLAN port membership vectors

VEPA Device Port Settings

Example VDP Configuration Scenarios

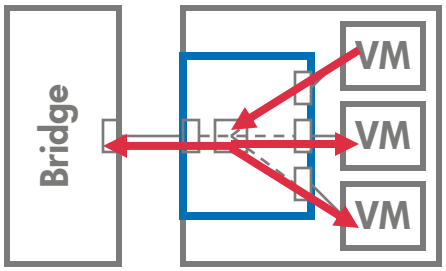
Settings	No VLAN Tag	Force Priority	VM has 3 VIDs	VM has 1 VID	private VLAN
MAC addresses	one+	one+	one+	one+	one+
Acceptable frame types					
– Only VLAN tagged			X		
– Untagged, Pri tagged	X	X		X	X
– All frames					
PVID	1	1	--	c	c
(Egress) VLAN IDs	1	1	a, b, c	c	c, d
Ingress VID Filtering	On	On	On	On	On
Priority Setting					
– Default value	n	n	n	n	n
– Set to default	False	True	T/F	T/F	T/F
Ingress MAC Filtering	T	T	T	T	T

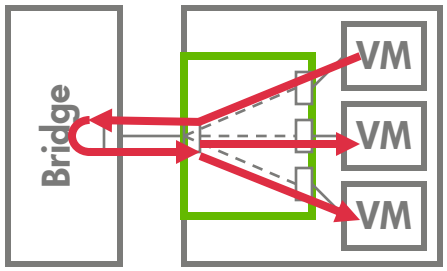
Summary

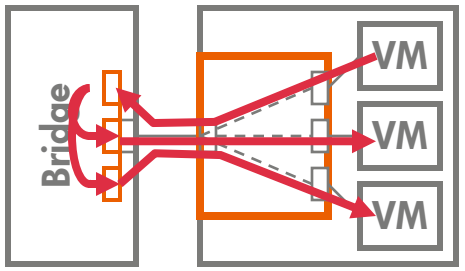
Approach Comparison

Area	VEB	Tagless VEPA	Tagged
Key Elements	Station VEB SW VEB	Station VEPA SW VEPA Expander Optional Portlets	Station VEPA + tag processing SW VEPA Expander Requires Virtual Switch Ports
Station-side Learning	Static, NIC-driven address table Special treatment of promiscuous ports	Static, NIC-driven address table (used on Ingress) Special treatment of promiscuous ports	No MAC address learning in VEPA (Uses static tag address table)
Station-side Forwarding (in)	Standard via use of static address table	Based on static address table	Based on static tag-to-port table
Bridge-side Learning	Standard	Standard	Standard + (must be aware of virtual ports)
Bridge Ingress Forwarding	Standard	Requires 'turn-around' mode	Requires 'turn-around' mode (tied to virtual bridge ports)
Frame Replication	Station-side replication	Station-side replication	Bridge-side replication (or station-side with extensions)
QoS	Set per VF (?) Single set of ETS queues	Set per VF Single set of ETS queues	Set per Virtual Switch Port Single set of ETS queues
Statistics	Limited (station-side collection)	Limited (station-side collection) ++	Limited (station-side collection) ++
ACLs	Limited	ACLs per 'portlet'	ACLs per virtual switch port
# of VMs	Nearly unlimited (via vswitch)	Nearly unlimited (via expander)	Determined by number of virtual bridge ports
Bridge traffic monitoring	Limited	Full	Full
Private VLAN Support	No	Yes	Yes

Summary of Possible Technical Approaches







multicast behavior

Virtual Ethernet Bridge (VEB)

uses MAC+VID to steer frames

- Emulates 802.1 Bridge
- Limited controls
- Managed by station
- Works with all existing bridges
- No changes to existing frame format.
- Open-ended changes to NIC

Tag-less VEPA

uses MAC+VID to steer frames

- Extends 802.1 Bridge
- Advanced controls
- Managed by bridge
- Works with many existing bridges
- No changes to existing frame format.
- Limits NIC changes

Tagged

uses new tag to steer frames

- Extends 802.1 Bridge
- Advanced controls
- Managed by bridge
- Works with few or no existing bridges
- Changes to existing frame format.
- Limits NIC changes