Edge Virtual Bridging: A potential simplification

Joe Pelissier

new-pelissier-EVBSimplification-0709

Where we are…

- The EVB group has been meeting weekly and has been discussion a variety of issues related to data center Ethernet deployments
- Two technologies have been discussed at length:
  - VEPA provides greater visibility and control over embedded bridges and potentially augments their functionality
  - VNTag removes bridges (and much of their associated management costs) from the network that are primarily performing aggregation functions
- These technologies address different problems in data center deployments
  - However, both of these technologies rely on forwarding a frame to a “controlling bridge” from which the frame may be forwarded back to the originating device (VEPA or Port Extender)

The problem…

- **For Port Extenders to correctly operate, the Controlling bridge requires knowledge of the PE's ingress port and the ability to explicitly indicate the PE's egress port(s)**
  - **Both are required at egress for proper multicast pruning**
  - **Various approaches have been discussed to eliminate the need for this knowledge or supply it implicitly**
    - For various technical and/or practical reasons, none were sufficient to promote migration from proprietary to standards based solutions
- **VEPA does not require this indication for proper operation**
  - **However, having such an indication does provide VEPA with additional capability**
- **The VEPA controlling bridge function may be implemented in most bridges without hardware modification**
  - **Providing the ingress/egress indications would require hardware modification in most cases**

A few bad paths…

- **Just do both…**
  - **Requires embedded devices to operate in two different modes with multiple hypervisor and OS implementations**
    - The test and verification matrix becomes impractical
- **Do one or the other…**
  - **Doing one does not address the problem set of the other**
  - **Just gets us back to the approach above, only worse**
    - One problem set get solved by proprietary solutions
- **Do neither**
  - **Worst case of all of the above**
  - **Both problems are solved by independent proprietary solutions**

More than any time in history mankind faces a crossroads. One path leads to despair and utter hopelessness, the other to total extinction. Let us pray that we have the wisdom to choose correctly.
– Woody Allen

Goals

- From a VEPA point of view:

  Enable VEPA using currently deployed controlling bridge hardware

  Non-goal: Enable VEPA in the middle of the network

  Non-goal: Produce a device that is significantly less complex than existing VEBs

- From the NIC/VEB/IV point of view:

  Reduce modes of operation

- From a Port Extender point of view:

  Enable Port Extenders at both the edge and in the middle of the network

  Produce a device that is significantly less complex than existing VEBs

  Produce a standard that provides equivalent functionality to the VNTag proposal

  Non-goal: Drive VNTag verbatim through the standards

  Non-goal: Eliminate the need for VEBs (or VEPAs)

  Non-goal: Ensure PEs work with existing CB hardware

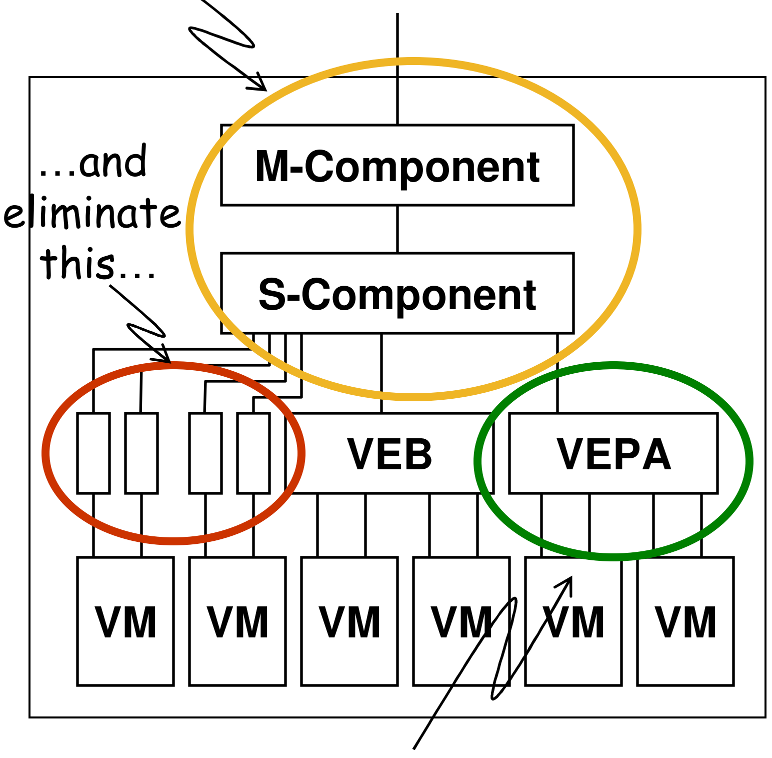
A potential path…

- **Tweak the VEPA requirements such that a VEPA provides the functionality of a Port Extender**
  - **An edge device may be a VEPA or a Port Extender**
  - **If an edge device is a VEPA, there is no point in having a “Port Extender” mode**
- **Requirements to achieve this:**
  - **An ability to provide an indication of ingress port**
  - **An ability to process the egress port indication (which may be a single port or a pointer to a list of ports)**
    - Or provide equivalent egress functionality (this is key!)

A simplified approach

- **An edge device that supports both VEPA and PE modes would look something like this ->**
    - **M & S components add little (or no) value to VEPA southbound**
    - **M & S components not necessarily required southbound for PE (could use VEPA forwarding tables)**
        - But does provide value in some cases
    - **Device processes two tag formats (M & S)**
    - **Northbound, STag provides only a VEPA indication, not a VM indication**
- **We'll examine how this might be accomplished in three steps:**
    - **Northbound path (VEPA -> CB)**
    - **Southbound path without replication (CB - > VEPA)**
    - **Southbound path with replication (CB -> VEPA)**

We can potentially simplify this…



…and eliminate this…

…by tweaking this

Heading North

From VEPA to Controlling Bridge

Breaking down a VEPA

- **The VEPA portion of the device looks something like this ->**

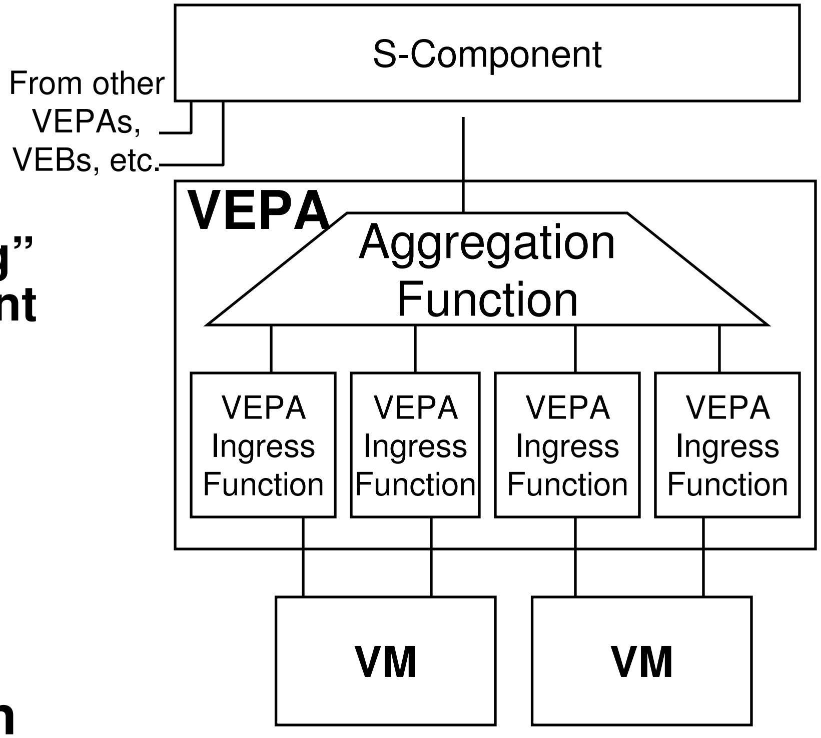
  The M-Component has been omitted since it does not perform any function northbound

  If the VEPA is attached to a “STag” capable CB, then the S-Component adds a tag that indicates the individual VEPA that sourced the frame

  If not, the S-Component simply aggregates the frames

- **However, an S-Component also performs an aggregation function**

  This creates an interesting possibility…



A Layered VEPA

- **This provides almost the same functionality, except:**

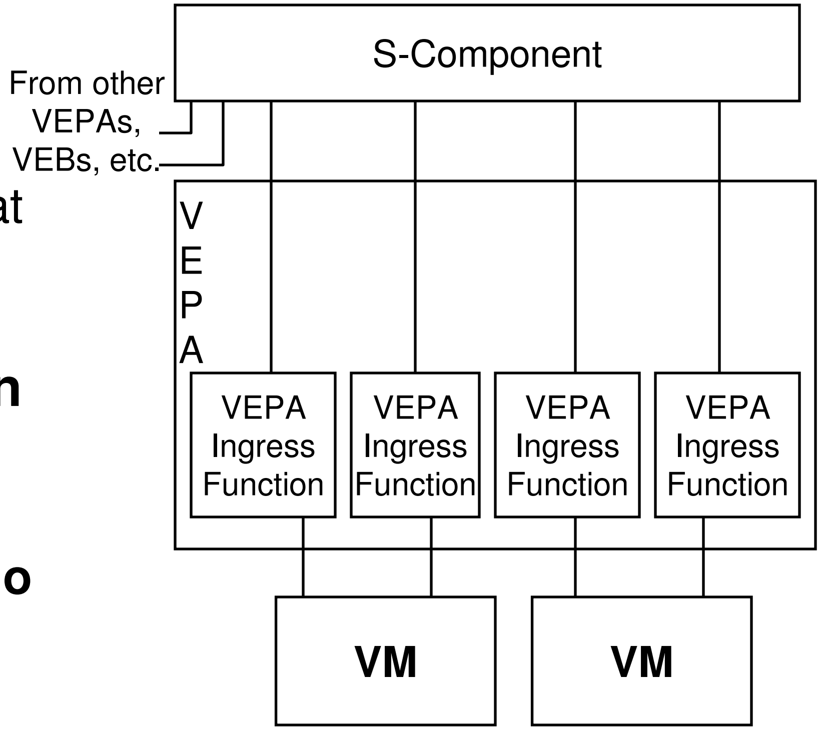
  The STag (if present) provides an indication of the VM, not just the VEPA

  Of course, we can make sure that a VM -> VEPA mapping is provided

- **Also note that one valid operation of the VEPA ingress is to do nothing**

  i.e. member of all VLAN groups, no ACLs, etc.

- **This VEPA function now does everything VEPA and provides PE capability (at least it does northbound)**

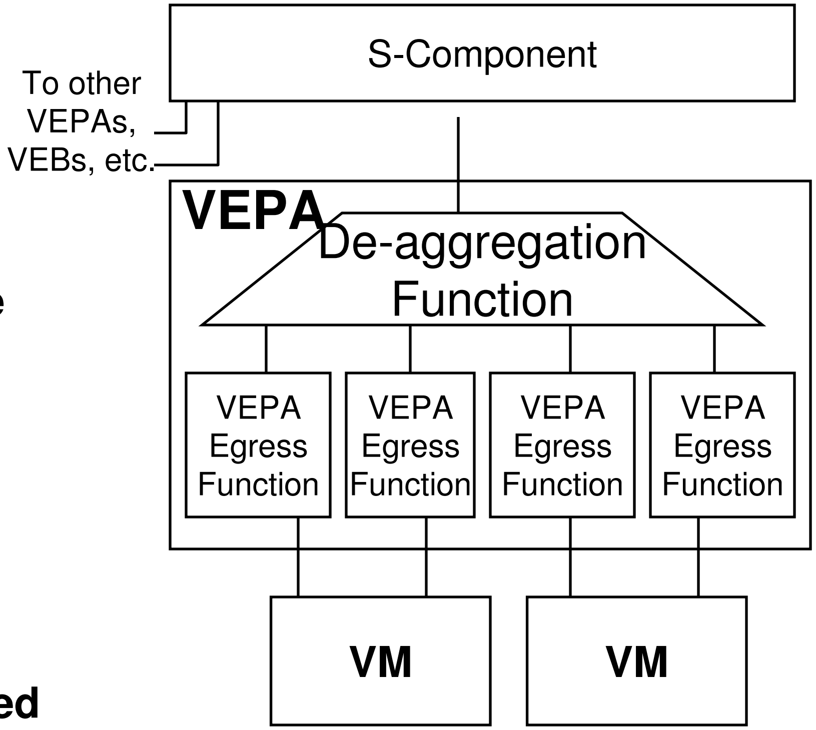


Heading South

From Controlling Bridge to VEPA without replication

Breaking down a VEPA (again)

- **The VEPA portion of the device looks something like this ->**
  - **The M-Component has been omitted since it does not perform any function southbound without replication**
  - **If the VEPA is attached to a "STag" capable CB, then the S-Component removes the STag and forwards to the appropriate VEPA**
    - If not, the S-Component simply forwards the frame to a given VEPA
- **This time, we cannot replace the De-aggregation function with the S-Component**
  - **The De-aggregation function is required to support the case of a non-STag capable Controlling Bridge**
- **However, we could provide both…**



A Layered VEPA

- In this model, the S-Componet is enabled if attached to an STag capable bridge

  Otherwise, De-aggregation is enabled

- This dual mode is *not* required

  De-aggregation provides the same behavior *(at least in theory)*

- However, when the STag is available, an implementation *may* find it advantageous to use it

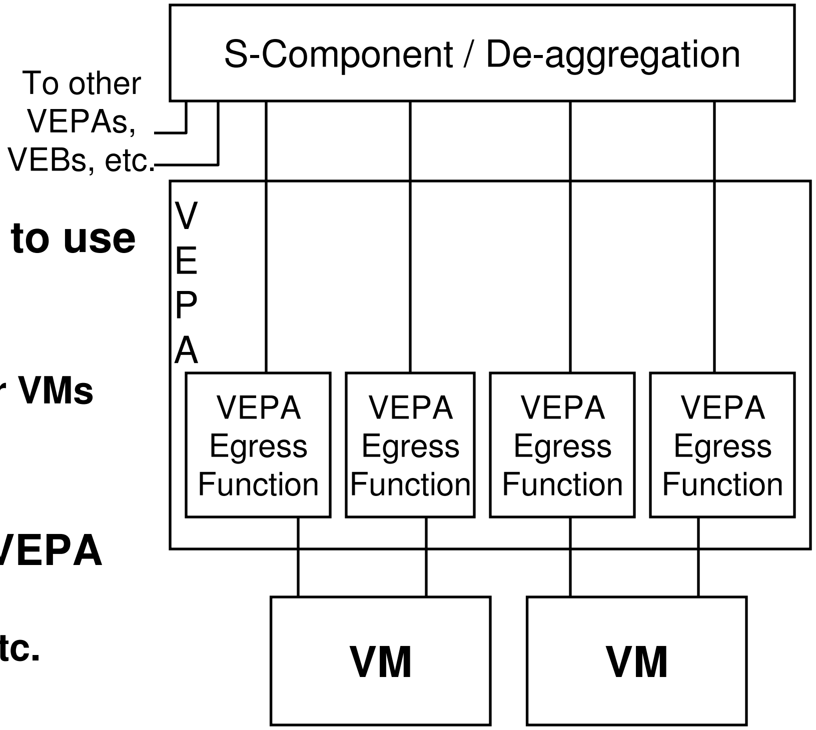
  Reduces address table space

  Provides learning capability (i.e. support for VMs operating in “promiscuous mode”

- No point in prohibiting this use
- Also note that one valid operation of the VEPA egress is to do nothing

  i.e. member of all VLAN groups, no ACLs, etc.

- With or without the S-Component, the functionality of both a VEPA and PE is provided

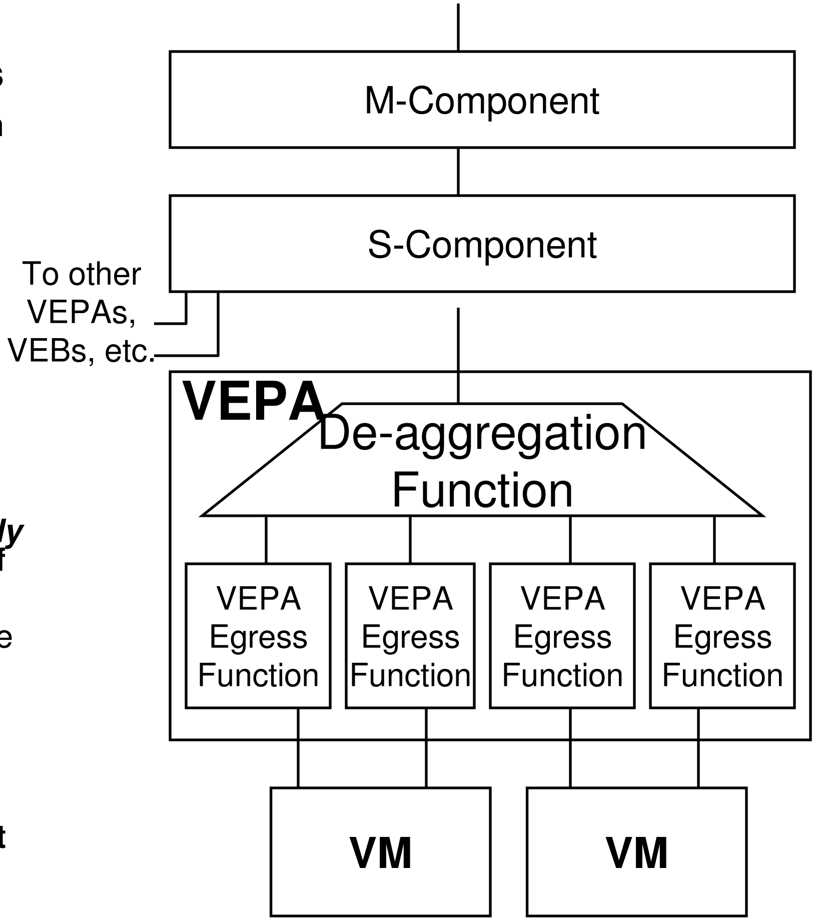


Heading South Again (and again, and again)

From Controlling Bridge to VEPA with replication

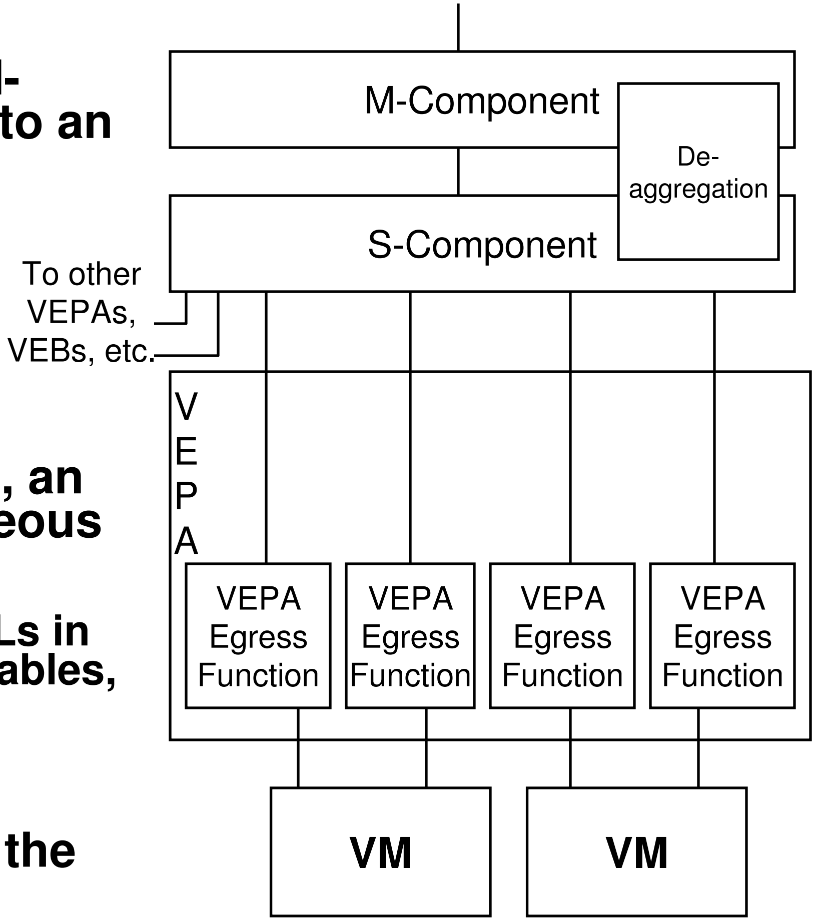
Breaking down a VEPA (again)

- **Replication is required for a variety of functions**
  - **VEPAs perform this function in the De-aggregation block for multicast**
  - **PEs need it for flooding, port mirroring, and multicast**
- **The VEPA portion of the device looks something like this ->**
  - **The M-Component is not required in the VEPA case since it performs replication based on MAC address (and never needs to flood since it has a priori knowledge of all MAC addresses, unless "promiscuous mode" is supported.**
  - **In the case of PEs, the M-Component *architecturally* interprets the M-Tag and creates multiple copies of the frame with appropriate STags**
    - The S-Component then forwards the frames to the appropriate ports
- **We cannot replace the De-aggregation function with the S-Component / M – Component combination**
  - **The De-aggregation function is required to support the case of a non-STag capable Controlling Bridge**
- **However, we could provide both…**



A Layered VEPA

- **In this model, the S-Componet and M-Components are enabled if attached to an STag/MTag capable bridge**
  - **Otherwise, De-aggregation is enabled**
- **This dual mode is *not* required**
  - **De-aggregation provides the same behavior (*at least in theory)***
- **However, when the MTag is available, an implementation *may* find it advantageous to use it**
  - **Enables external egress multicast ACLs in the CB, reduces space in forwarding tables, etc.**
- **No point in prohibiting this use**
- **Also note that one valid operation of the VEPA egress is to do nothing**
  - **i.e. member of all VLAN groups, no ACLs, etc.**
- **Either way, the functionality of both a VEPA and PE is provided**



Replication Observations

- Note that the M-Component and S-Component layering is architecturally elegant, but kind of a pain to implement

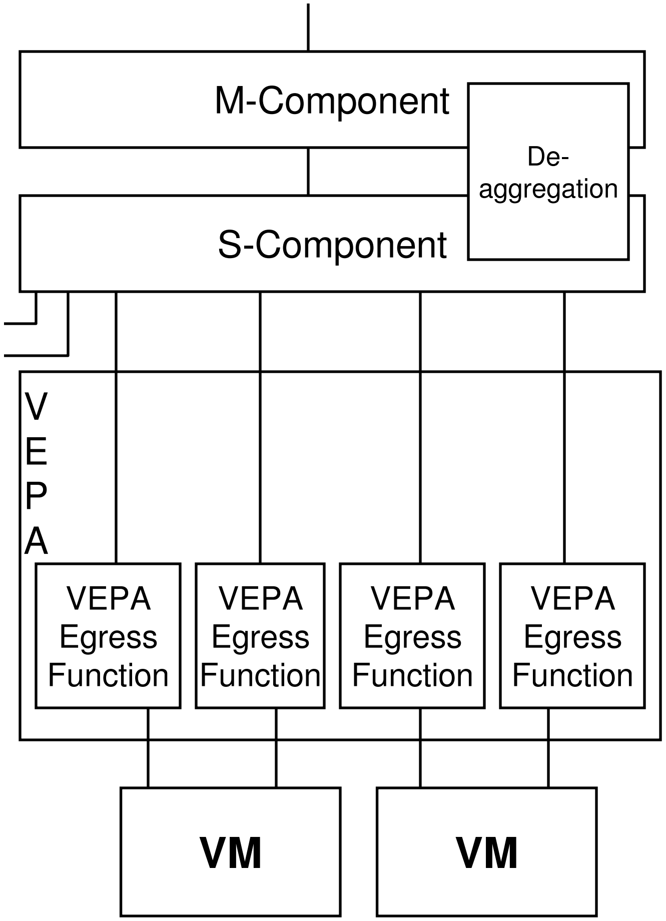
   Optional for VEPAs

   Required for PEs

- Frames may come in with two different tag formats (STag or MTag)

   CB must produce these two different formats

- A tag could be created that performs both functions (which I'll call an LTag)



Replication Observations

- If attached to an LTag capable CB, the L-Component function is enabled

  Otherwise, the De-aggregation function is enabled

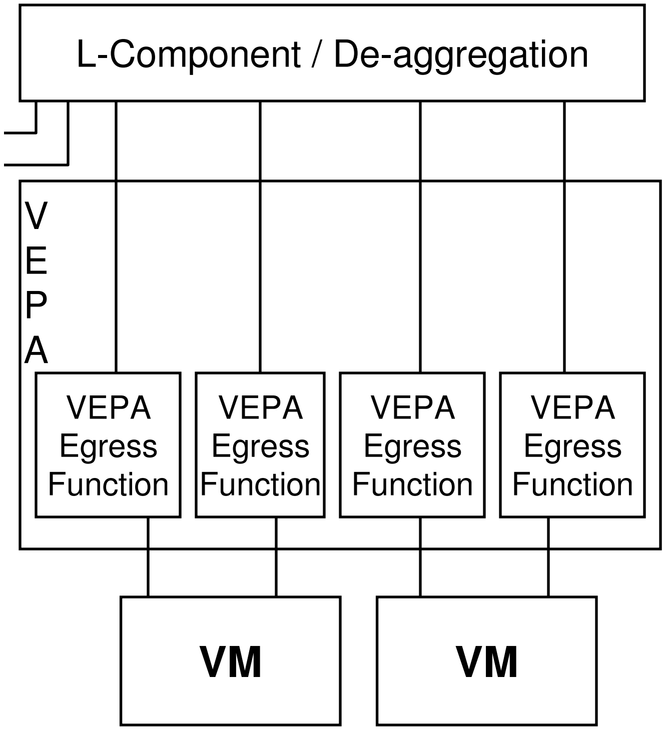
- The LTag contains:

  An indication of the source port

  An indication of the destination port or port list

  An indication of whether the destination is a port or a pointer to a list of ports

- The L-Component removes the LTag and forwards to the appropriate VEPA Egress Function or Functions



Multi-channel support

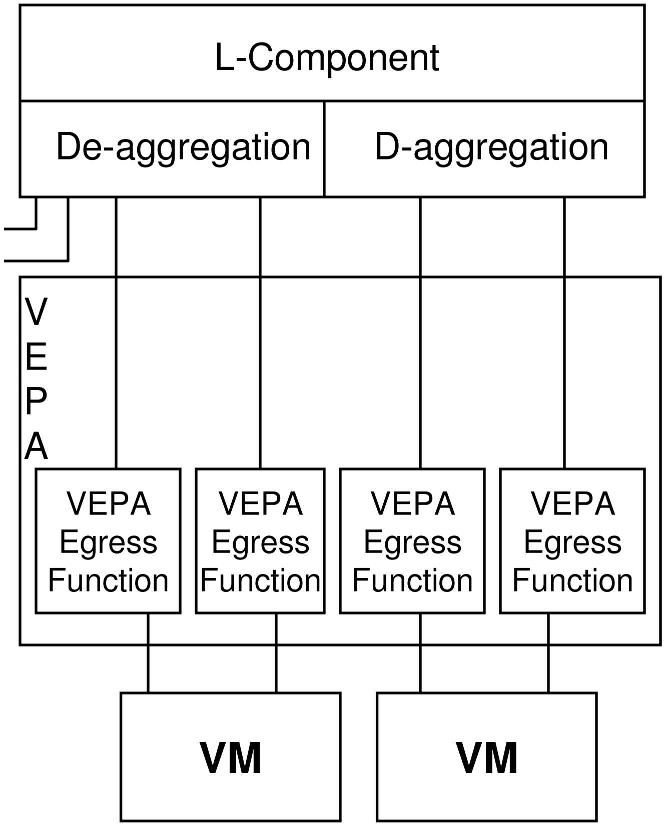
- **In the original architecture, an STag could be used to route a frame to a particular VEPA**

  Then the VEPA de-aggregation function performs the replication based on MAC/VLAN

- **A similar approach is possible here:**

  An LTag is used to route to a given De-aggregation function

  The De-aggregation function performs replication



Summary

This is surprisingly easy!

Summary

- An edge device can be:

  A Port Extender

  A VEPA

  A VEB

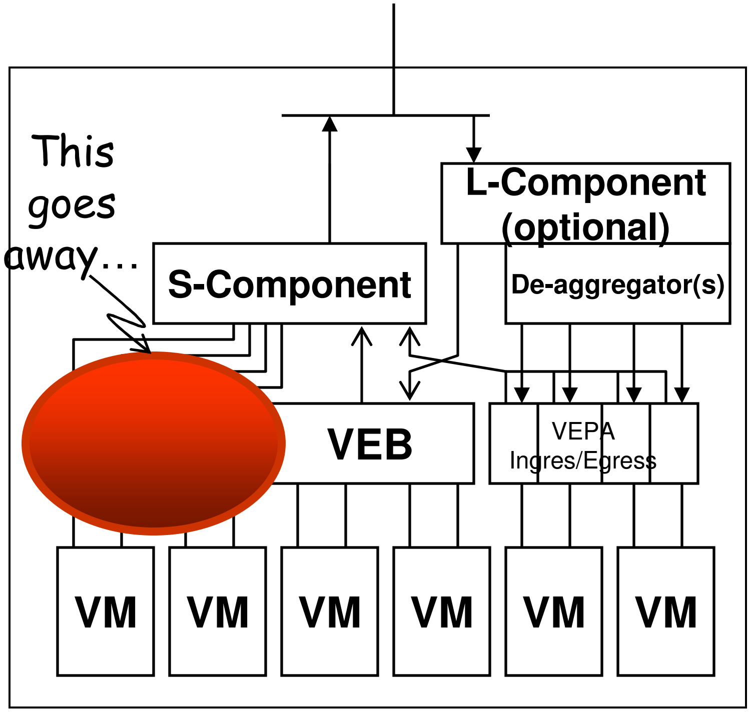
  Or, a combination

- Only one functional change to VEPA is required to eliminate any need for an edge device to support both modes:

  Provide an ingress port indication rather than an ingress VEPA indication in the STag

  The proposed architecture provides this

- Optionally, an implementation may choose to provide an L-Component southbound



Summary

- **Discovery and Operation**
  - **The edge device discovers if the CB is “STag capable”**
    - If so, the tagging function in the S-Component is enabled
    - If not, the tagging function in the S-Component is disabled
  - **The CB (potentially through intervening PEs) discovers if the edge device is “LTag capable”**
    - If so, the CB and PEs forward the LTag
    - If not, the device immediately upstream from the edge (CB or PE) removes the LTag

Summary

- What exactly is a Port Extender?

  Northbound it’s an S-Component

  Southbound it’s an L-Component

Thoughts on PARs

- **We need to define:**
  - **Definition of PE operation**
    - Requires S-Component Extension, L (or M) component definition, hairpin mode (?)
  - **Definition of VEPA ingress/egress operation**
    - Requires S-Component Extension, (IMHO) L (or M) component definition, hairpin mode
  - **Extension to S-Component:**
    - allow it to not tag in certain cases (when a VEPA is attached to a non-STag aware bridge and when an STag is already present)
  - **Definition of "hairpin mode" operation**
    - The "hairpin mode" being discussed in RCSI may be more appropriate for PE operation
    - Dependant upon definition of VEPA and PE
      - IMHO, hairpin mode is dangerous enough that we should not start a project to define it until we have consensus on what is going to attach to it
- **Essentially, everything depends on everything else**
  - **Potentially have a single "Bridge Extension" PAR to cover all of it?**

Questions and Thoughts?

Thank You!