



# Port Extension support for Generic Bridge Types

Joe Pelissier

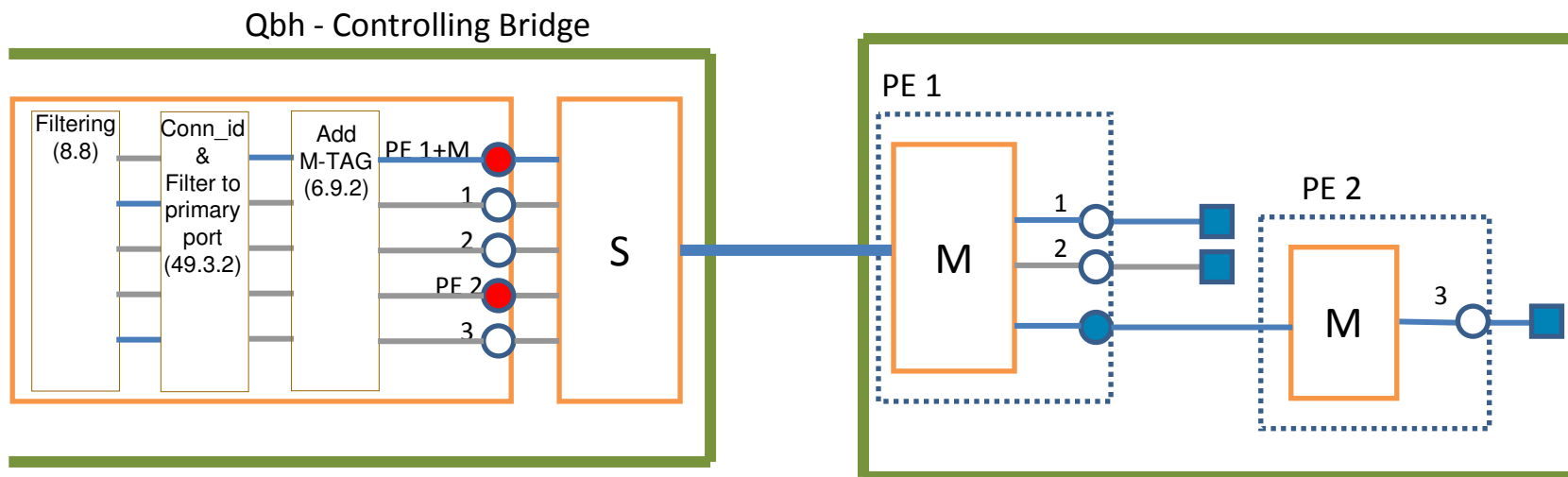
bh-pelissier-genericbridges-0910

# Goals

- **Enable Port Extension to accommodate generic bridge types (i.e. in addition to C-VLAN components)**
- **Simplify overall architecture**
  - Align M-component more closely to S-component**
  - E.g. see [dcb-congdon-QbgQbhAlign-0710-v1.pdf](#)**

# Current Proposed Architecture

- Relay function performs filtering (8.8)
- M-channel Registration Entries (8.8.9) produce a connection\_identifier (49.3.2)
- Additional filtering (49.3.2) to the Filter Group Primary Port
- EISS (6.9.2) adds an M-TAG
- Connects to a SVMC that adds S-TAGs for unicast but not for multicast.



# A Cleaner Approach

- **Consolidate as much new functionality as possible into the M-component**

**It was proposed in dcb-congdon-QbgQbhAlign-0710-v1 that the M-component should perform all of the additional filtering and insertion of M-Tags instead of the C-VLAN component**

Removing this functionality from the C-VLAN component also reduces the impact of supporting Port Extension with other bridge components (although not technically necessary)

**Difficulty is defining how the M-Comp is to detect and filter multiple copies of a multicast frame.**

# A Cleaner Approach (2)

- **Observations**

**In the current architecture, the S-VLAN Mapping Component is attached to the C-VLAN component with an internal “LAN”**

**Such internal “LANs” can pass parameters in unspecified ways**

**If the C-VLAN component maintains the M-channel Registration Entries, it can create the connection-identifier**

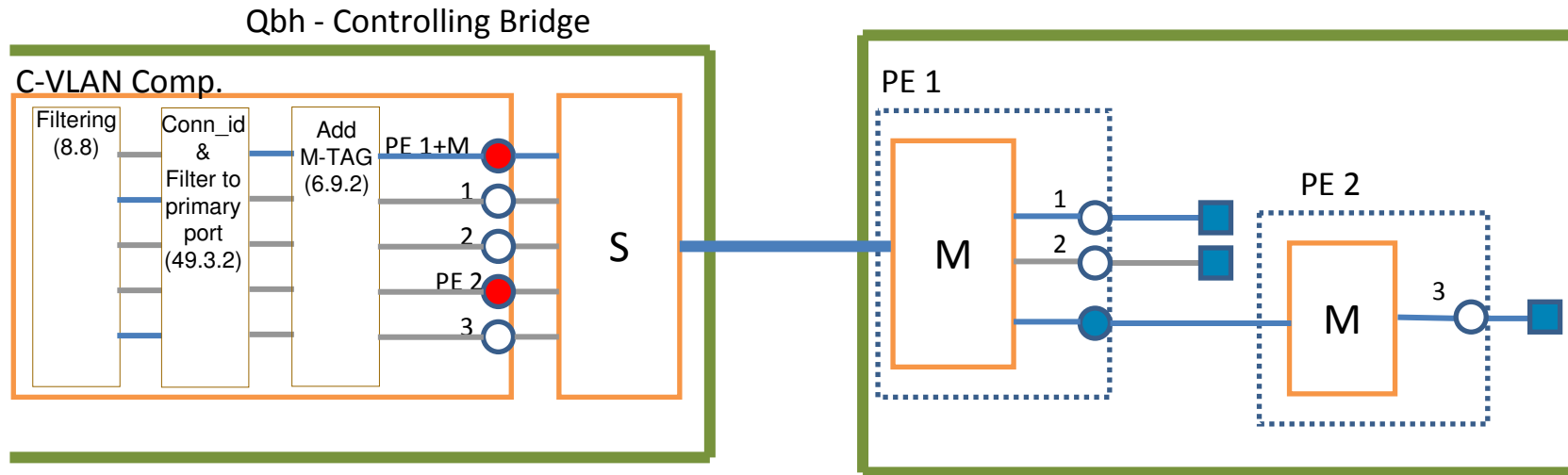
**If the connection\_identifier is passed on the internal “LAN” to an M-component instead of a SVMC, the M-component can perform:**

- The M-TAG insertion function described in 6.9.2

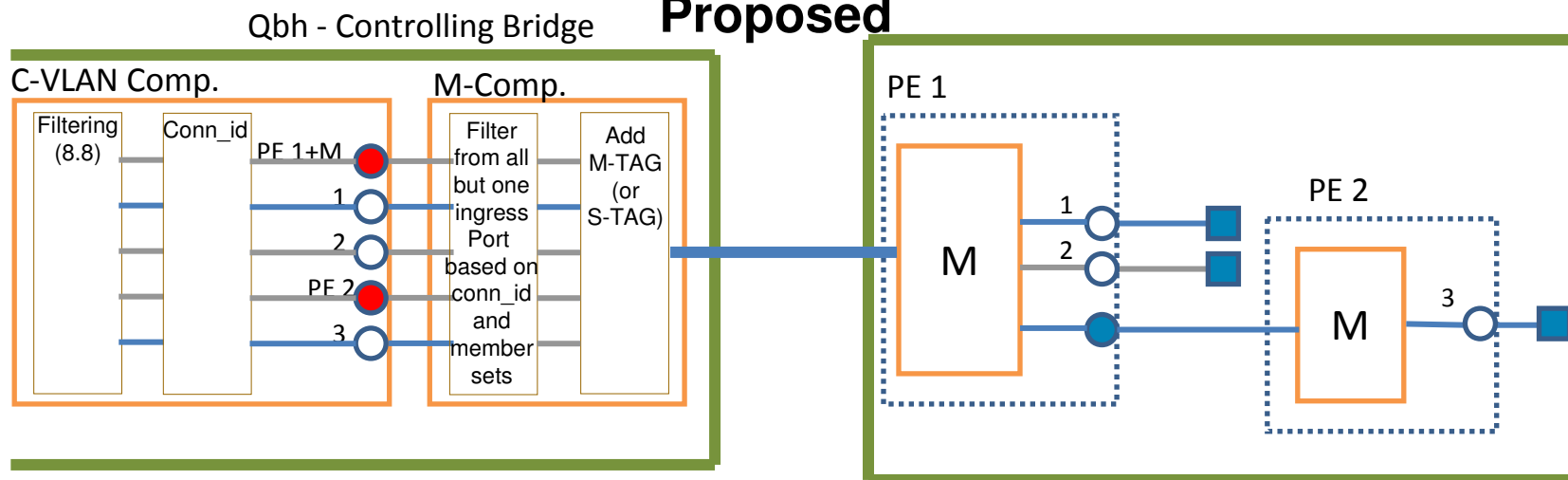
- The additional filtering required in 49.3.2

In addition, the entire concept of the Filter Group Primary Port is eliminated

# A Cleaner Approach (3)



Existing



Proposed

# Supporting Generic Bridge Types

- **The C-VLAN component in the current figures in the specification is replaced with a generic bridge component (representing a C-VLAN, S-VLAN, and any other relevant components).**
- **Any generic component wishing the use PE would need to support the one additional filtering entry (i.e. the M-channel Registration Entries) to generate the connection\_identifier.**
- **An M-component is instantiated connected to the generic bridge component by multiple internal “LANs”.**

# Supporting Generic Bridge Types (2)

- **What about the S-TAG?**

**If the M-Component is attached to an S-Component, then we have multiple devices generating an S-TAG.**

- **Option 1:**

**Create a new tag (e.g. a unicast or U-TAG). This would be identical to the S-TAG except that it has a different Ethertype**

- **Option 2:**

**Use our existing new tag, the M-TAG.**

Thus, the M-component generates a single tag.

Any other S-TAGs (e.g. for provider bridging or multichannel) are passed transparently through the M-component



# Supporting Generic Bridge Types (3)

- Using an M-TAG only, the dual relay and special routing in the EISS is eliminated
- The M-comp becomes identical to a reduced function S-comp except:

It processes / emits an M-TAG instead of an S-TAG

The PE CSP is supported

Source port filtering is supported

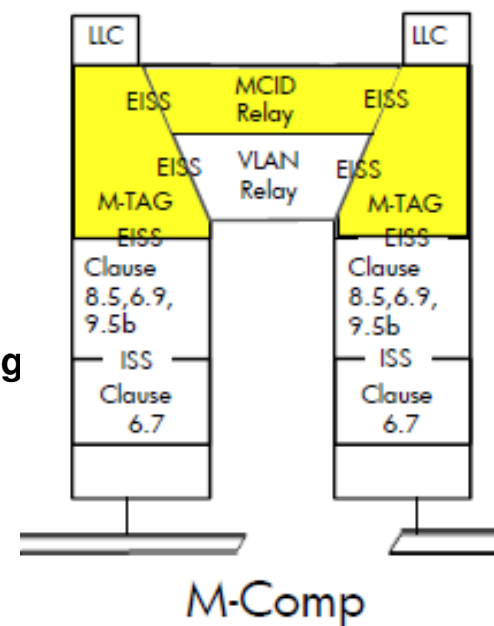
Some terms change:

Service VLAN becomes M-channel

SVID becomes MCID

The MCID space is bigger than the SVID space

- All tagging, untagging, and forwarding is accomplished through configuration of the member and untagged sets
- Reduced function:
  - The static and dynamic MAC tables are not required
  - STP not required

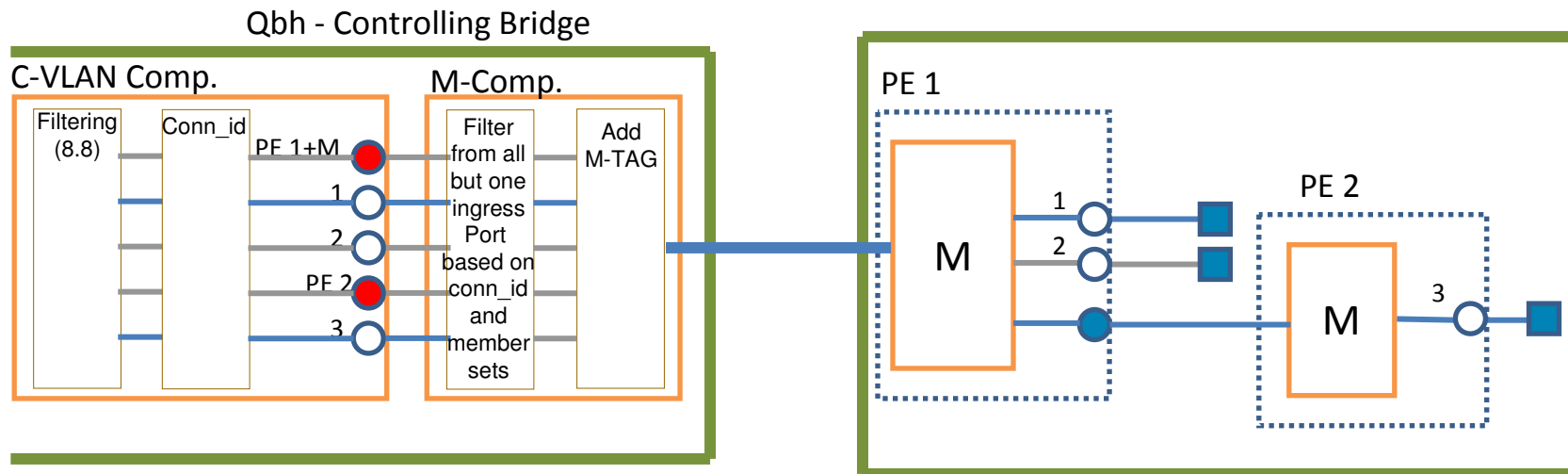


The yellow portions are eliminated by using a single tag.

# Supporting Generic Bridge Types (4)

- **M-component always emits an M-Tag**
- **If connection\_identifier present, the M-Tag is multicast (MCID obtained from connection\_identifier)**
- **Otherwise, the M-Tag is unicast (MCID is the ingress port default MCID)**

**Incoming frames (eastbound in this figure) are never multicast, therefore the connection\_identifier is never present.**

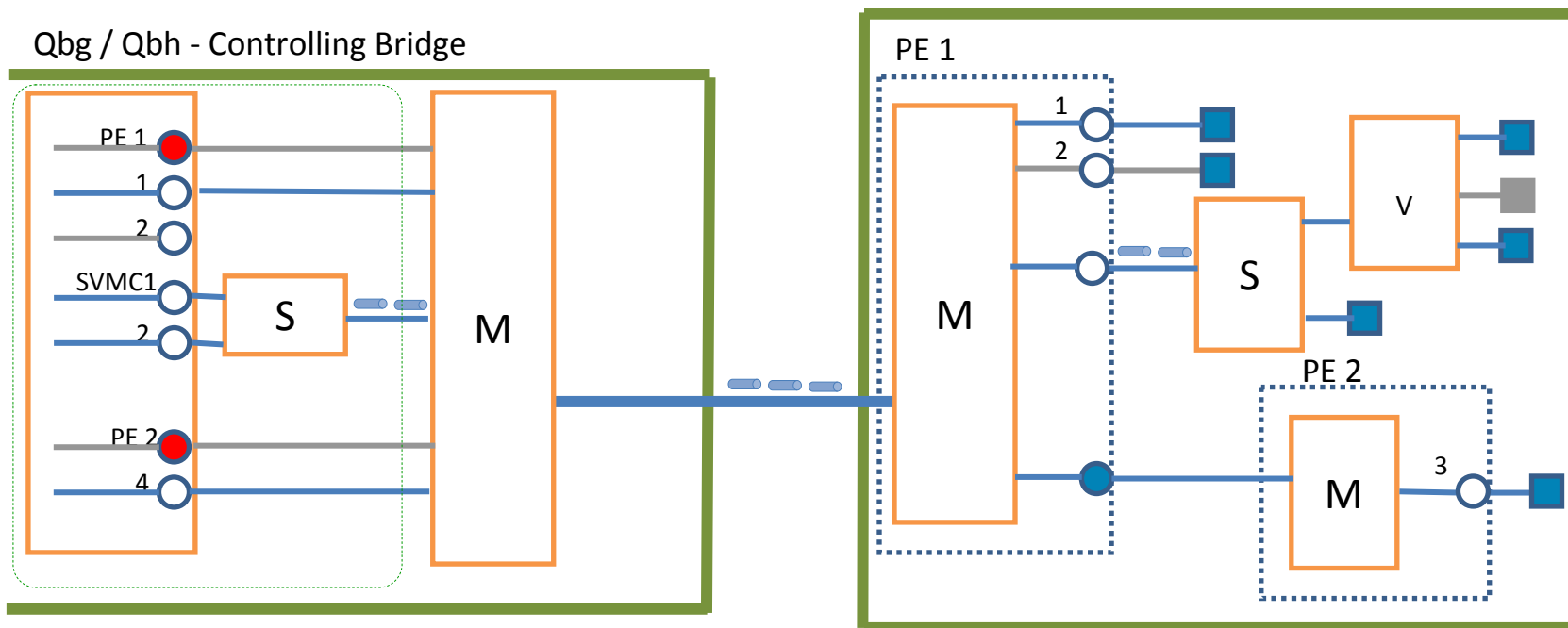


# Supporting Multichannel

- A bridge that supports multichannel is just a generic bridge type from the point of view of the M-component
- S-channels are point-to-point, so replication of a frame from a SVMC is never required

Therefore, SVMC does not provide `connection_identifier` on internal “LAN”

S-channels and S-TAGs pass through M-component transparently



# Summary

- **Provides PE support to generic bridge types including those that support Multichannel**
- **Reduces new behavior in existing components**
- **Reduces new behavior in the M-component**
- **Maintains one component – one tag behavior**
- **Provides better modularity between the functionality of Qbg and Qbh**
  - e.g. Port Extenders do not require knowledge of multichannel operation
- **Overall cleaner and easier to understand specification**

---

Thank You!