Problem list for P802.1Qbz / P802.11ak point-to-point model

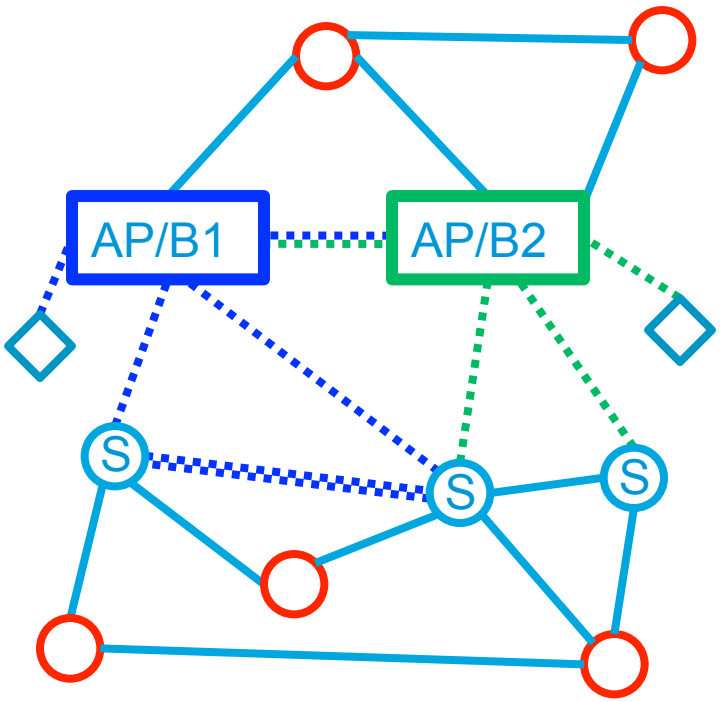
Norman Finn
December, 2012

Version 2

Introduction

- This presentation is available at: http://www.ieee802.org/1/files/public/docs2012/bz-nfinn-pt-to-pt-problem-list-1112-v02.pdf
- There are a number of issues that need to be solved when integrating IEEE 802.11 media into the core of the network. Some are of concern primarily to P801.Qbz, and some concern primarily 802.11ak. Some may be solved by either project.
- This deck concentrates on problems that (this author thinks) could be solved by P802.11ak. (Whether the eventual solution will be in P802.1Qbz, P802.11ak, or not specified.)
- There are often many possible ways to solve each problem.
- This deck primarily concentrates on the problem statement, not the solution.
- **This deck assumes the use of the point-to-point model for 802.11 integration into the network, not the emulated LAN or emulated bridge model.**

Set of point-to-point links

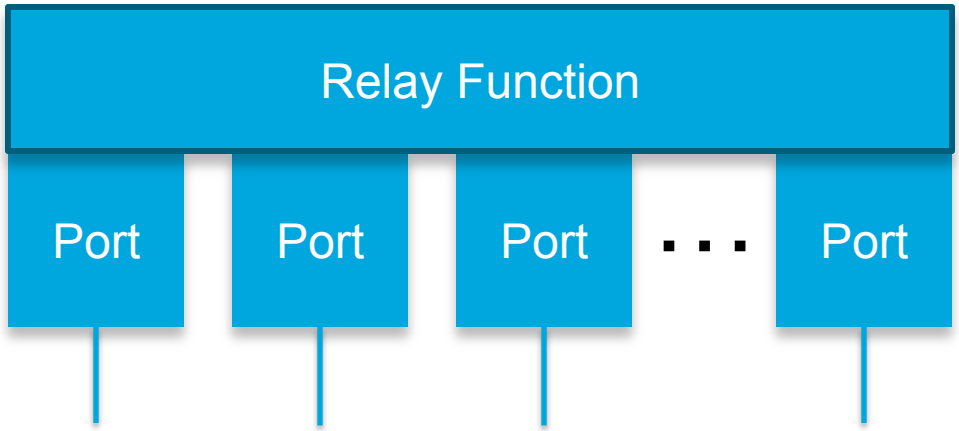


- The Access Points and their co-resident bridging functions become integrated AP bridges (AP/Bs).
- Devices with non-AP station capability(ies) and wired connections become “non-AP station bridges” (S).
- Of course, not all stations are bridges. (The diamonds are non-bridge non-AP stations.)
- Each wireless connection appears, to the bridge functions of the system, to be a separate instance of the MAC service.

MAC service instances

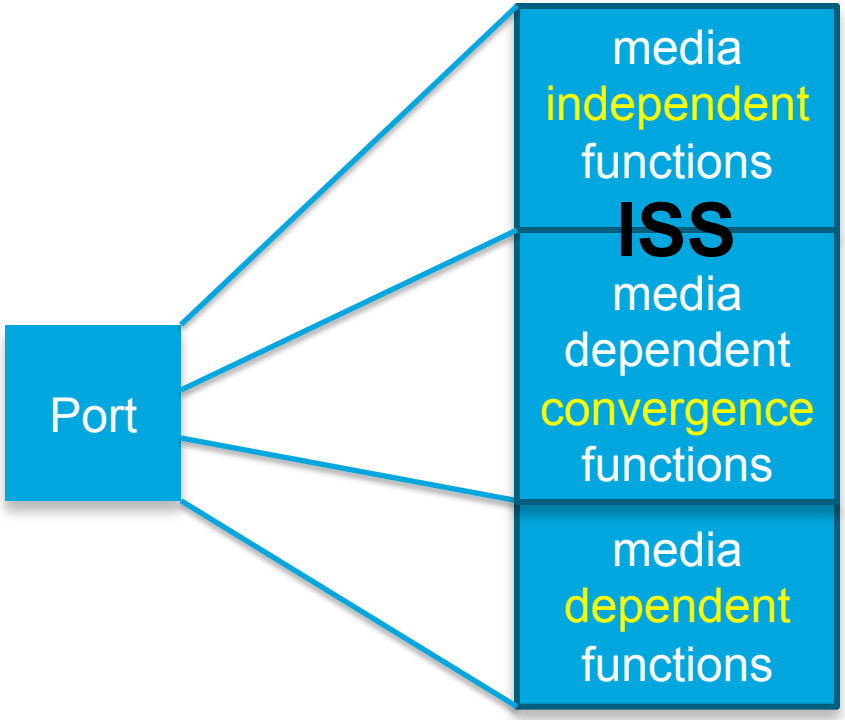
MAC service instances

- In terms of data forwarding, an 802.1Q bridge is a **Relay Function** attached to some number of **Ports**, with each Port offering an instance of the **MAC service**.



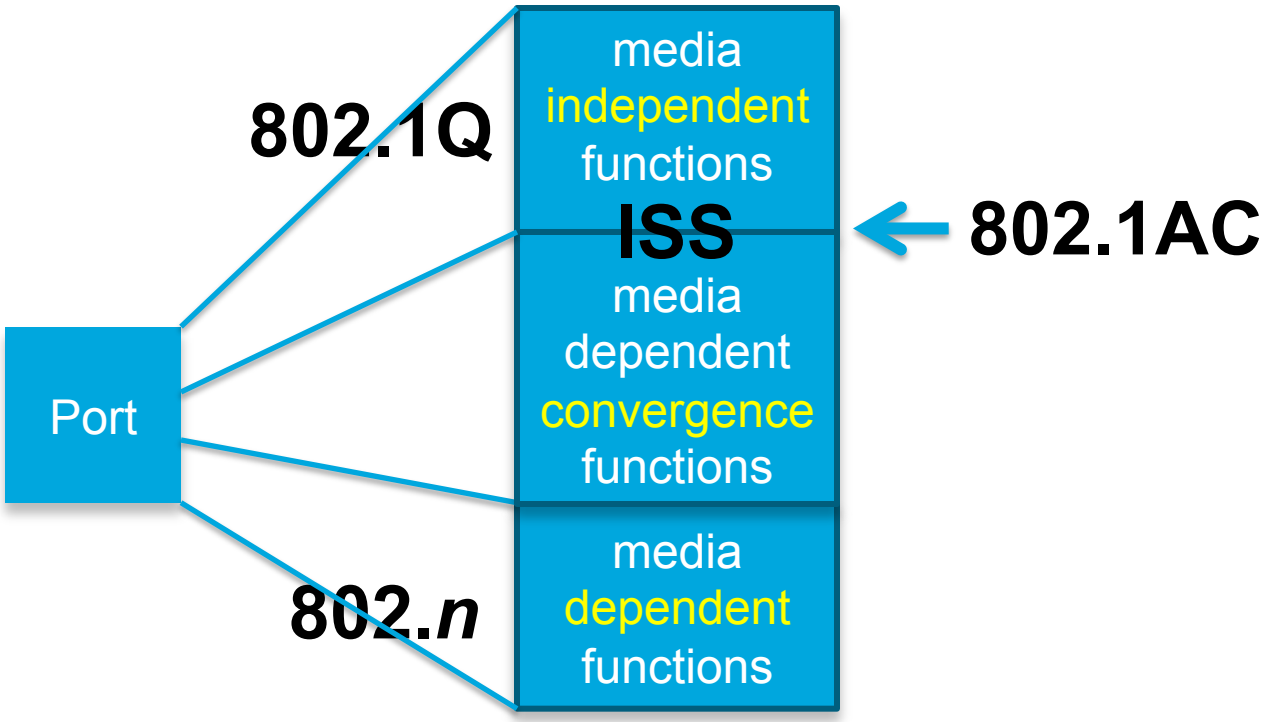
MAC service instances

- A Port can be expanded into:
  - The media-**independent** functions;
  - The media-dependent **convergence** functions; and
  - The media-**dependent** functions, that vary according to the medium.

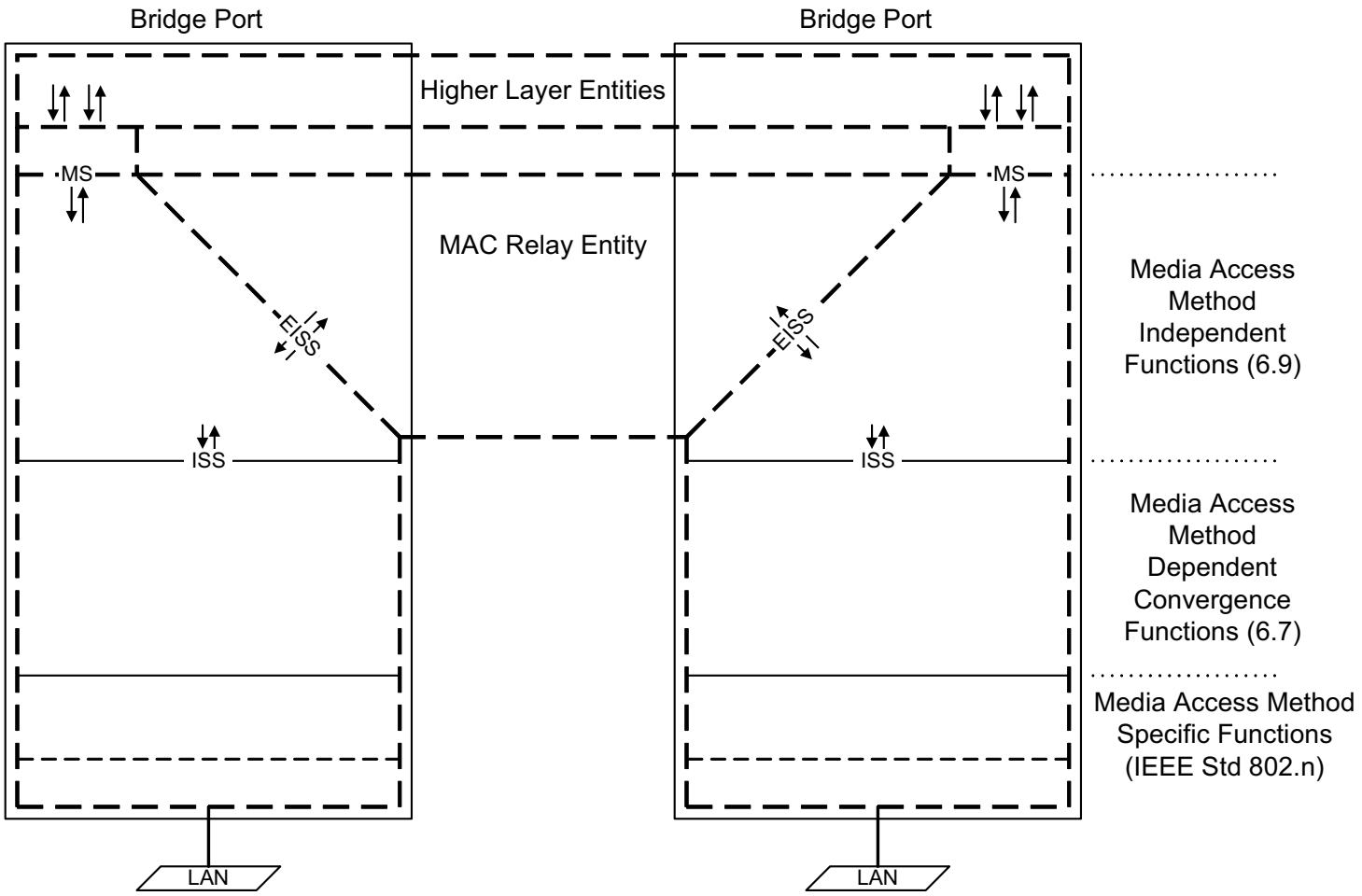


MAC service instances

- The demarcation between 802.1Q and a specific medium is a MAC Service Access Point (**MSAP**) offering the Intermediate Sublayer Service (**ISS**), defined in 802.1AC-2012.



IEEE Std 802.1Q-2011 Figure 8-2



NOTE—The notation "IEEE Std 802.n" in this figure indicates that the specifications for these functions can be found in the relevant standard for the media access method concerned; for example, n would be 3 (IEEE Std 802.3) in the case of Ethernet.

Figure 8-2—VLAN-aware Bridge architecture

IEEE Std 802.11-2011 Figure 5-1

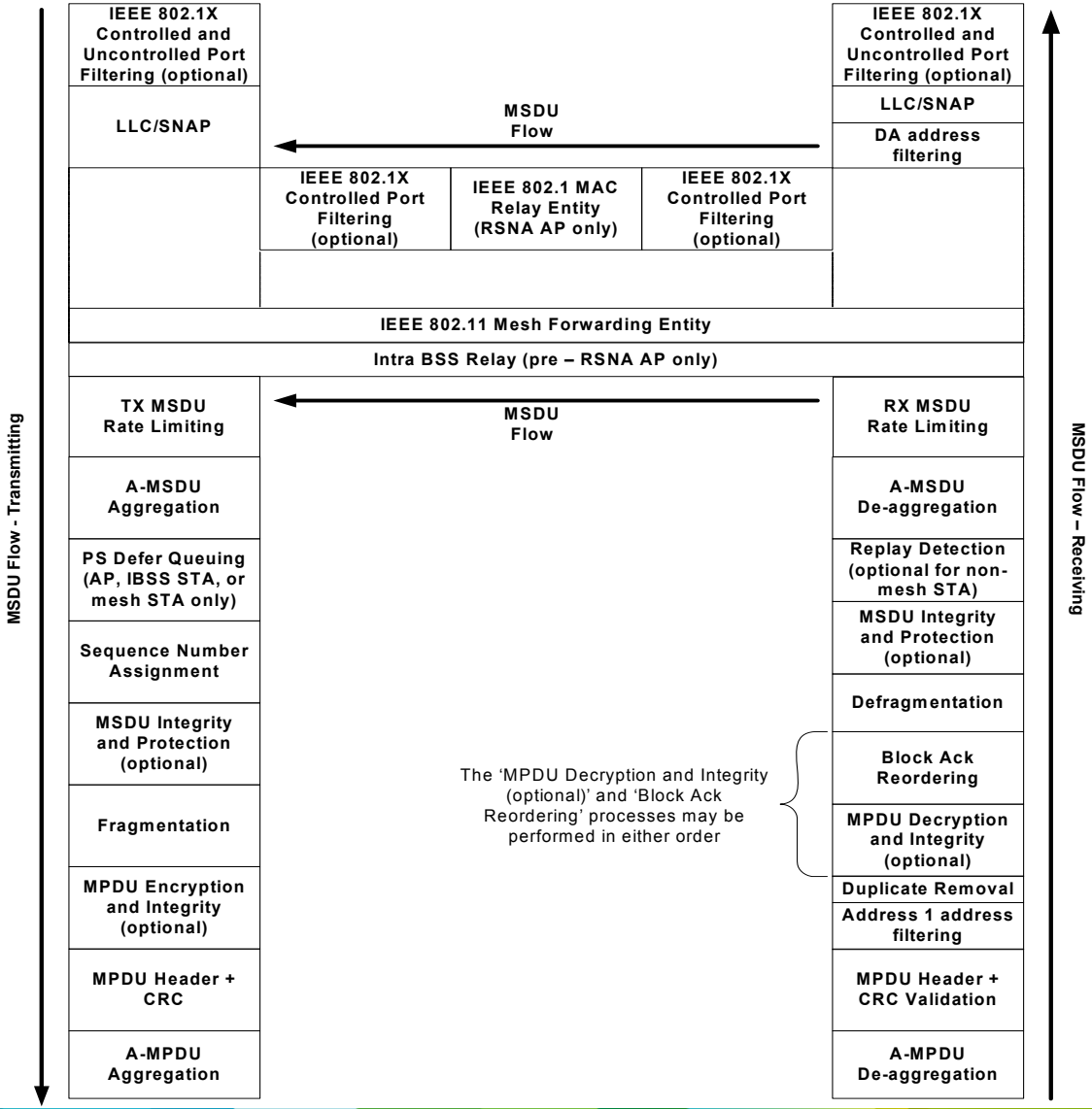


Figure 5-1—MAC data plane architecture

MAC service instances

- 802.11-2012 Figure 5-1 seems to be largely compatible with the corresponding 802.1Q-2012 Figure 8-2.
- The only real differences, at least at the diagram level, seem to be that:
  - The 802.1Q “Media Access Method Independent” and “Media Access Method Dependent Convergence” functions are in the place occupied, in 802.11, by an optional “IEEE 802.1X Controlled Port Filtering” function and a blank box.
  - The Bridge model seems to require replicating frames to separate ports, rather than sending one copy to multiple “ports”.
  - A similar .1X CPF is also placed above the LLC in the 802.11 diagram.
- The “Higher Layer Entities” in 802.1Q are not present in 802.11 because the 802.11 diagram is limited to the data plane.

MAC service instances

- Apparently, the differences regarding 802.1X filtering is merely a matter of representing the controlled and uncontrolled ports described in IEEE 802.1AE-2006. These differences do not appear to be a problem at this point.
- Apparently, the individual ports in 802.11 Figure 5-1 each represent a connection to a separate non-AP station. If this is true, it represents exactly the “point to point model”.
- **If we apply 802.11-2012 Figure 5-1 to both Access Point stations and non-AP stations, it should not be difficult to reconcile the current 802.11 and 802.1Q standards with the point-to-point model.**

MAC service instances

- To morph 802.11 Fig. 5-1 to 802.1Q, either:
  1. Relabel the “IEEE 802.1 relay entity” to “IEEE 802.1 relay and media dependent functions”; or
  2. Show the media dependent and convergence functions on each side of the relay.
  3. Label the blank box at the “T” intersections “Media Access Method Independent Functions”.
- To morph 802.1Q Fig. 8-2 to 802.11:
  1. Substitute, for the “media dependent functions” of Fig. 8-2, the dual ingress/ egress stack of functions shown for each port in Fig. 5-1.
- And, of course, resolve the 802.1X filtering issue.
- It is possible that a closer examination of the 802.11 port stack will reveal further issues.

Reflection problem

Reflected frames: normal non-AP use

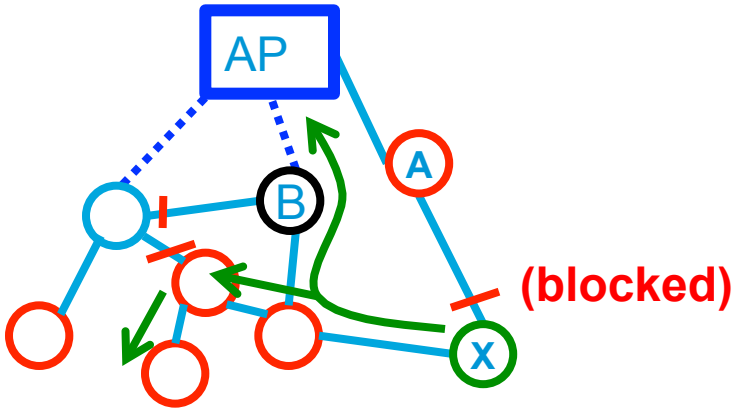
- A non-AP station uses its own MAC address as both the Ethernet source address and the transmitter address. The AP uses the Ethernet destination as the receiver address. Hence, three addresses are sufficient for both directions.
- A broadcast UP frame (non-AP station to AP):

- A broadcast DOWN frame (AP to non-AP station):

- If the station sees its own MAC address in the Ether Source, it discards the frame, else it passes it up to its MAC client.

Problem set-up

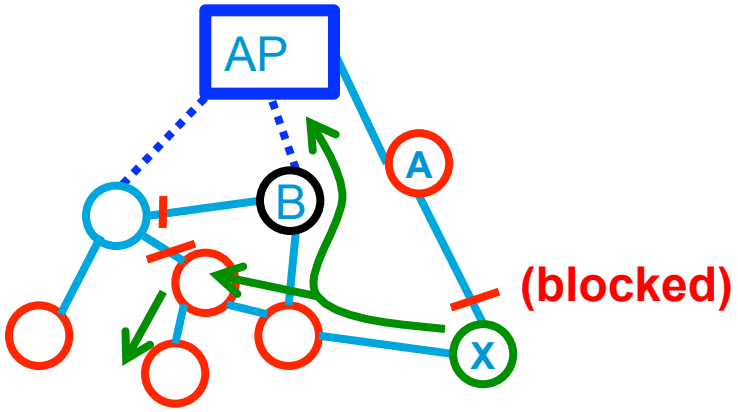
- Imagine a network where the AP, stations, and bridges are all running either the Spanning Tree Protocol or Shortest Path Bridging.
- Imagine a station X attached to bridge X, transmits a broadcast frame.
- Suppose that bridge “B” is a laptop with a Wi-Fi station and two 802.3 ports, and that we turn on a standard 802.1Q bridging function implemented in software.



Reflected frames: the problem for bridges

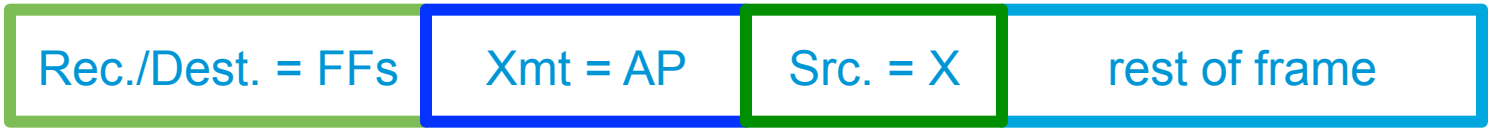
- **CASE 1:** Suppose a non-AP station/bridge ***B*** is forwarding data for attached wired device ***X***.
- Suppose ***X*** sends a frame (a broadcast, for example) up through bridge ***B***.

CASE 1

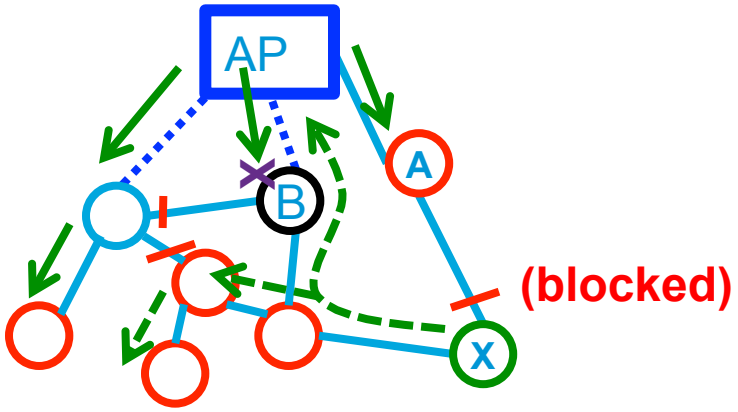


Reflected frames: the problem for bridges

- In a wired network, the Bridge replacing the AP would **not reflect** the frame back down to Bridge ***B***.
- In a Wi-Fi world, the AP **does reflect** the frame back down to all of the AP's stations, including ***X***, and Bridge ***B*** needs to **discard** the frame. (Its portion of the network has already seen it.)

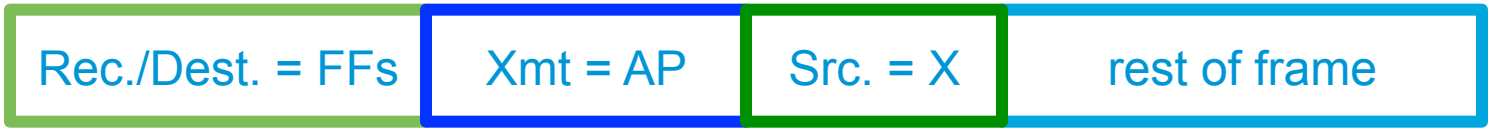


CASE 1
discard

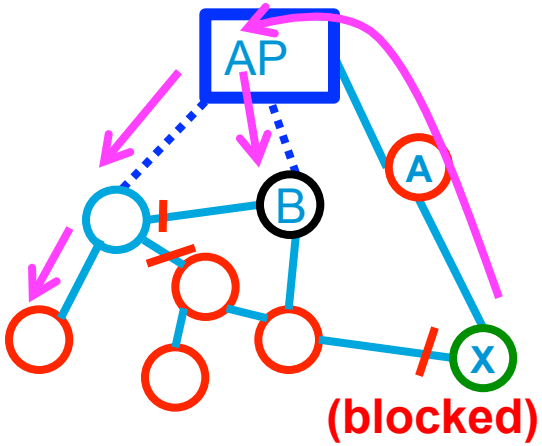


Reflected frames: the problem for bridges

- **CASE 2:** Suppose instead, that the spanning tree has changed, so that ***X*** has effectively moved, and transmits that same broadcast frame.
- The Access Point transmits the broadcast to all of its stations, including bridge ***B***.

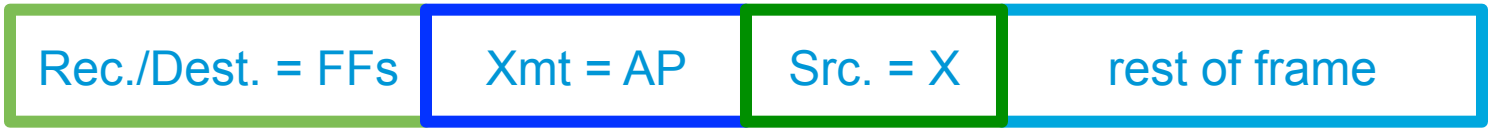


CASE 2

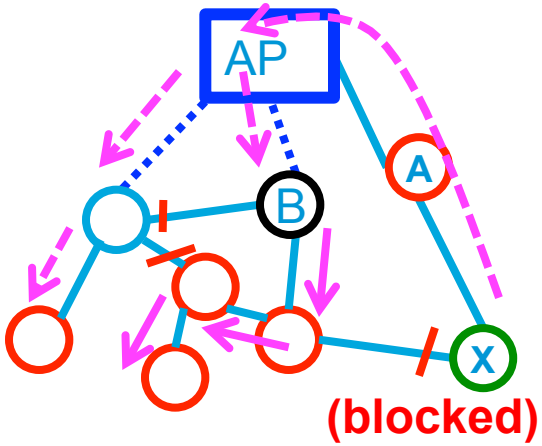


Reflected frames: the problem for bridges

- Bridge ***B*** **must relay** the frame down to the part of the network that hasn't seen it, yet.

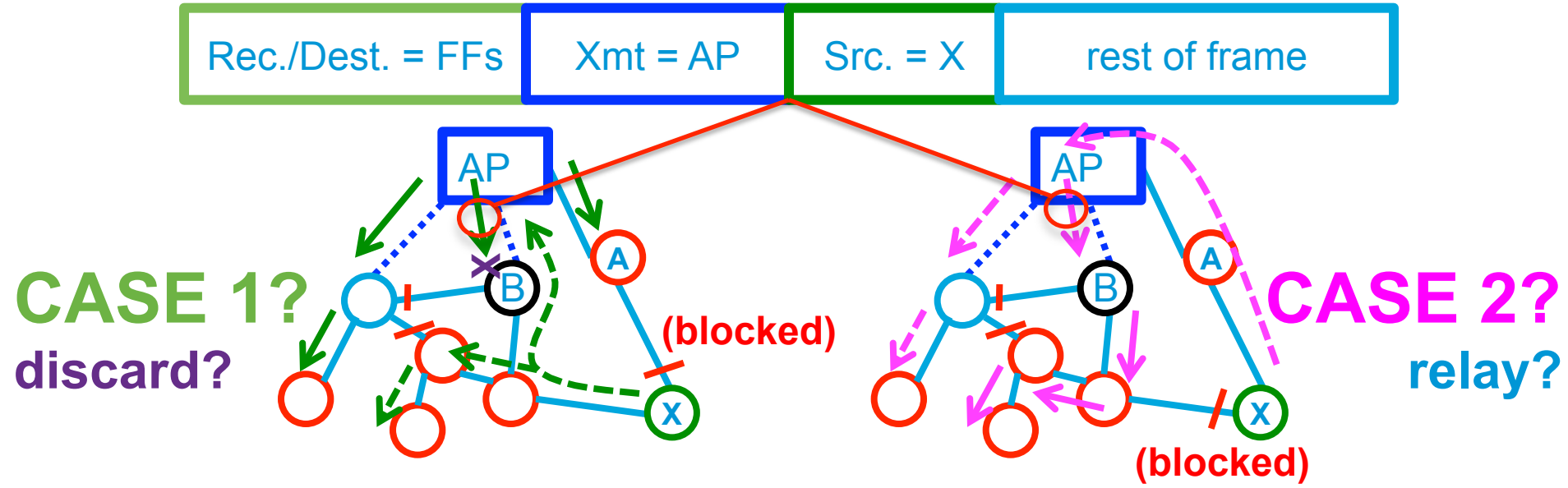


CASE 2
relay



Reflected frames: the problem for bridges

- The problem is that, **today**, Case 1 and Case 2 result in **exactly the same frame** sent from the AP to bridge ***B***.
- Bridge ***B*** **doesn't know** whether to **discard** the frame (the correct action in Case 1) or to **forward it and learn *X***'s new location (the correct action in Case 2); it cannot distinguish the two cases.



Two non-solutions

- **Question:** Why can't the bridge just discard frames based on the MAC addresses that it knows are "behind" it.
- **Answer:** If there were no wired connections below the bridge/stations, or closing the loop between the bridge/station and the AP, that would be possible. (It is, in fact, done today in a non-standard but common behavior.) But, in the general case, it is **only through learning** source addresses that the bridge "knows" anything about what is or is not behind it. The problem, here, is that there is nothing in the frame to tell the bridge whether to **apply** its already-learned knowledge or to **learn** new knowledge to apply, later.

Two non-solutions

- **Question:** Why can't the bridge just remember what frames were sent to the AP and discard them if and when they come back?
- **Answer:** Frames have different priorities. If the frame were reflected simultaneously with transmission (as in the original Fat Yellow Coax), the device could discard it easily. But, there can be an arbitrary time delay between the UP frame and the reflection. The bridge would have to store all UP frames in a content-addressable memory and look for matches. Not only is this very expensive, but frames can be lost, and duplicate frames can be legitimately sent, which further confuses this plan.

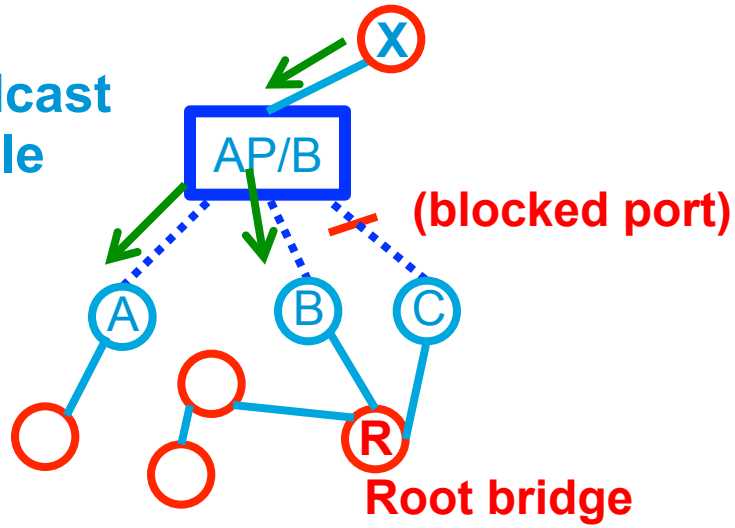
Additional constraints on any solution

- **Flooded unicasts:** It is not only frames with multicast or the broadcast addresses that may be reflected back down by the AP, or sent to all bridges by the AP. The AP must distribute frames sent to unicast addresses that are unknown to it to all station/bridges, but the station/bridge that sent it up to the AP (if any – maybe it came through the wire to the AP) must know to discard it.
- **Old stations:** We must know what existing non-AP stations will do with any new frame formats used. If one AP transmission can suffice for frame to be accepted or discarded by both the appropriate non-bridge stations and the appropriate bridge stations, that would be ideal. If two transmissions (in different formats) are required of the AP, we must be certain that existing stations, new simple stations, and new station/bridges, all will each pass only one of the frames to their respective MAC clients.

Multicast distribution

Multicast distribution

- Each device below is a bridge, wireless connections are treated as point-to-point links, and a broadcast frame is sent by bridge **X**.
- Suppose bridge **R** is the spanning tree root, so that one of the AP's "ports" is blocked.
- In the standard spanning tree protocol, bridge **C** does not know that the AP's link to it is blocked.
- **How does the AP forward the broadcast to A and B, but not to C, with a single transmission?**
- There are potential solutions to this problem in both the 802.1 and 802.11 spaces; there are tradeoffs to be explored.



Multicast distribution

- One solution would be to extend/modify MSTP and Shortest Path Bridging in P802.1Qbz to provide a handshake for bridge **C** to tell the AP/bridge that it knows the AP end of the link is blocked, so it is OK for the AP to send it to all; bridge **C** will discard it. (**This does not solve all of the multicast distribution problems**: see "Egress tagging".)
- A parallel solution is to send multiple unicasts to the bridges, at least until the handshake (if any) is done.
- A complete solution would be to provision a set of multicast Receive Addresses, in frames sent by the AP, to specify sets of bridge / stations. (In this case, "**A** and **B** but not **C**".)
  - This latter idea has its own problems – there are 2^N possible sets of destinations.
  - Perhaps we limit an AP to at most 24 bridge/stations (the number of bits available following the OUI in a MAC address), or define a protocol for distributing a mapping of vectors of stations to 24-bit IDs.
- Hopefully, someone has a better idea than any of the above.

VLAN tags

VLAN tags

- Any of an AP/bridge, a non-AP station bridge, or a VLAN-aware non-AP station can use VLAN tags.
- An AP/bridge or non-AP station bridge can be in the path of a VLAN for which it has no local customers or local use.
- Native 802.11 frames use the IEEE 802.2 LLC format.
- Therefore, adding a VLAN tag to a frame requires adding 10 bytes to the frame (8-byte SNAP encoding + 2 byte payload), instead of the 4 bytes (2-byte EtherType + 2 byte payload) in 802.3.
- Among bridges, a large fraction (typically, all) of the frames carry VLAN tags.

Unreliable links

Unreliable links

- Wireless media are inherently less dependable than wired media.
- The effective speed and availability of a given connection can vary over a short timescale.
- It is not acceptable to export volatile link conditions to the rest of the network; the result could easily be that the whole network becomes unstable.
- We should discuss heuristics for reporting link conditions, and explore the capabilities of the topology control protocols (MSTP and SPB) with respect to variable links.

Multicasts and Bridge Ports

Multicasts and bridge ports

- In an 802.1Q Bridge, a multicast frame received on one port is often replicated and transmitted on multiple separate ports.
- An Access Point can transmit a single frame to multiple non-AP stations. This seems to violate the Bridge model.
- 802.1 has discussed this problem in the past. My summary of these discussions is that, while some text needs to be added to 802.1Q, there is no fundamental problem, as will be explained in the following slides.

IEEE Std 802.1Q-2011 Figure 8-10

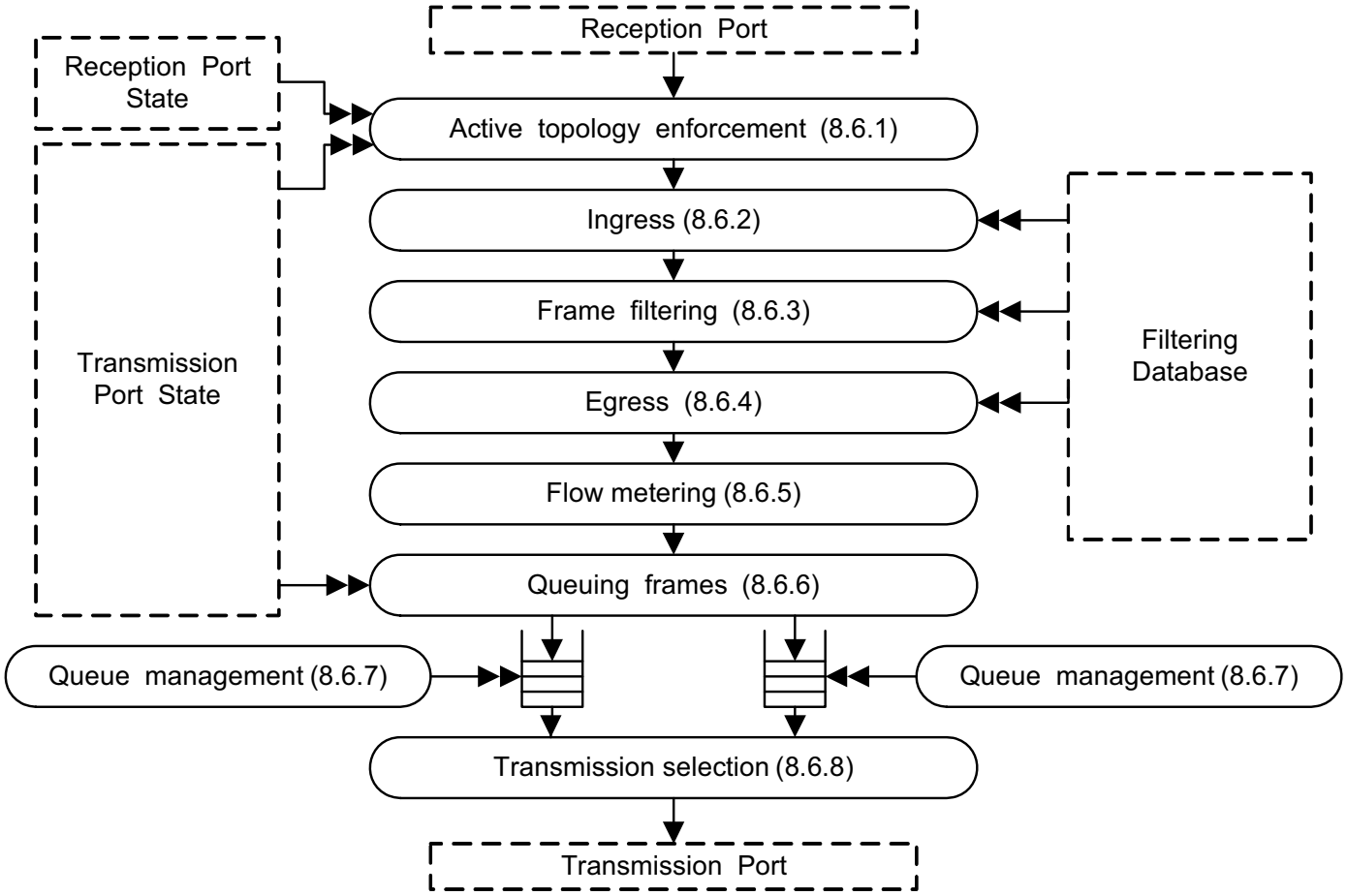


Figure 8-10—Forwarding Process functions

802.1Q forwarding process

- 8.6.1: Active topology enforcement.
  - Decide whether port state (blocking) allows frame to enter the relay.
  - Construct a list of all active (non-blocked) ports (not including the ingress port) and attach it to this frame.
- 8.6.2 Ingress.
  - Discard the frame if its VLAN tag is not correct.
  - Learn the frame's source MAC address.
- 8.6.3 Frame filtering
  - Use the Filtering Database to filter the port list. That is, eliminate as many ports from the output port list as possible, based on destination MAC address.
- 8.6.4 Egress
  - Discard frame from certain ports based on VLAN.

802.1Q forwarding process (continued)

- 8.6.5 Flow metering
  - Yellow/green/red marking according to input port
- 8.6.6 Queuing frames
  - Place frame in one or more queues, according to list of ports on which it is to be output.
  - Discard frames based on yellow/green/red marking.
- 8.6.8 Transmission selection
  - Select a frame to transmit on a given port.

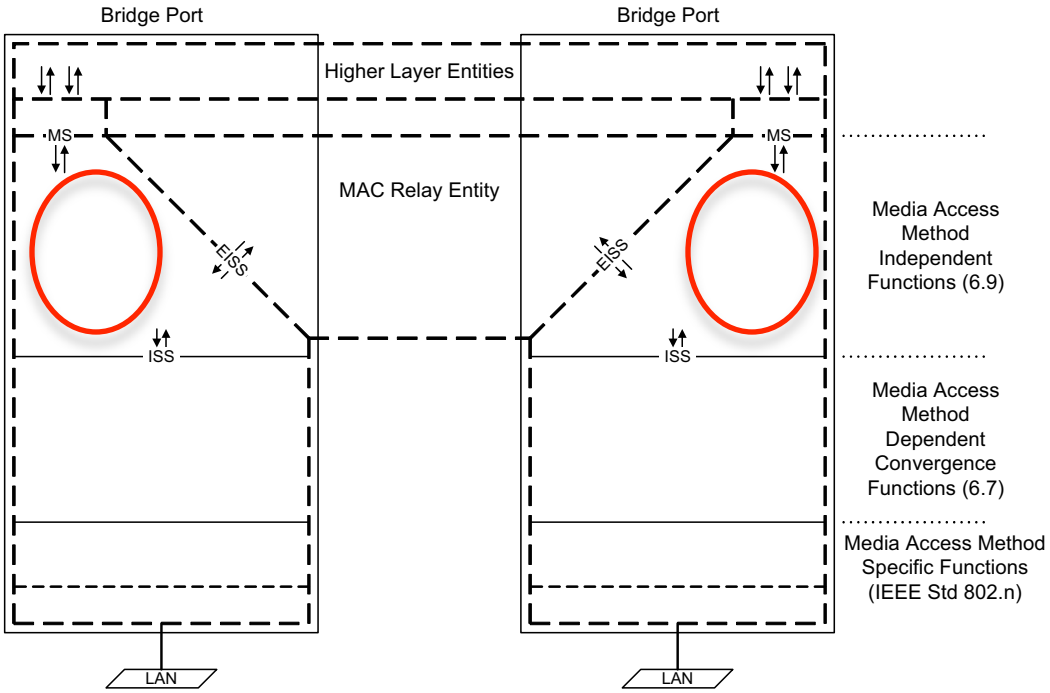
802.1Q frame forwarding

- One interesting point is that the Bridge does not replicate the frame; it constructs a list of ports on which the frame is to be transmitted.
- As far as the 802.1Q Relay Function is concerned, **the same identical frame is transmitted on all ports**.
- The differences between a wired Bridge and an Access Point can thus be buried in “8.6.6 Queuing frames”; that’s the only place it makes a difference, and that is exactly where the implementation issues are.
- So, it should be easy to modify 802.1Q to refer to 802.11 in clause 8.6.6 to clear up both the broadcast replication and queuing issues.
- (Well, almost. See “Egress tagging”, below.)

Egress tagging

802.1Q egress tagging

- Note that it is the “Media Access Method Independent Functions (6.9)” that interpret an incoming VLAN tag or generate an outgoing tag on a given port; this is outside the relay function, in the location in the 802.11 diagram occupied by an empty box.



NOTE—The notation “IEEE Std 802.n” in this figure indicates that the specifications for these functions can be found in the relevant standard for the media access method concerned; for example, n would be 3 (IEEE Std 802.3) in the case of Ethernet.

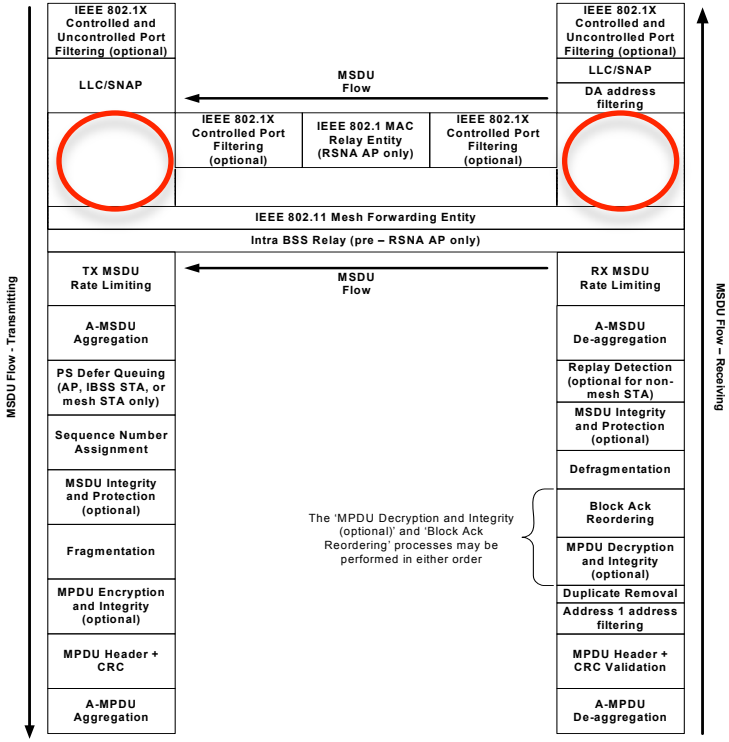
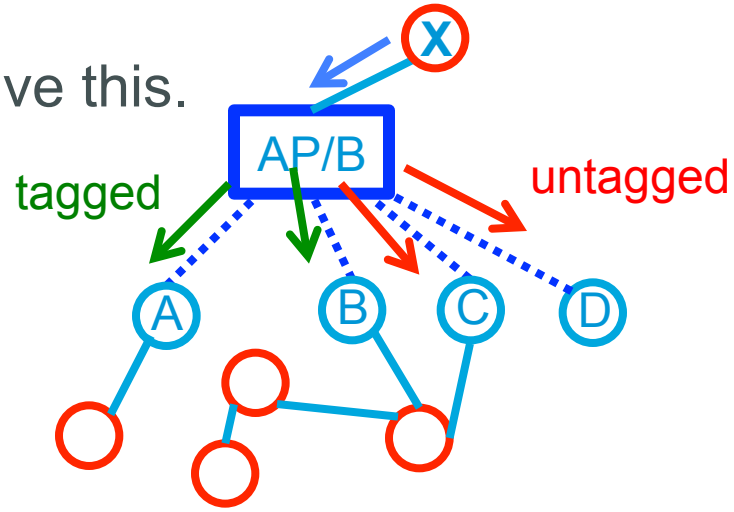


Figure 5-1—MAC data plane architecture

Figure 8-2—VLAN-aware Bridge architecture

802.1Q egress tagging

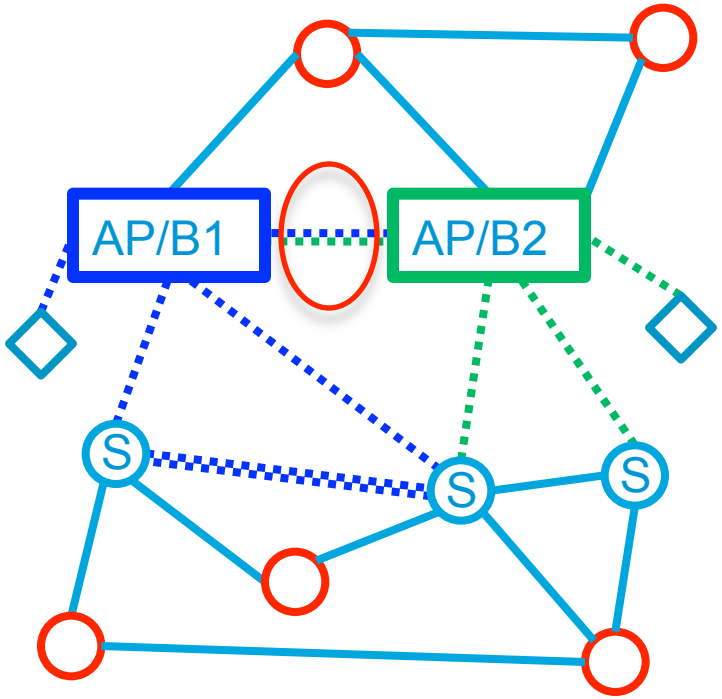
- Whether a frame is tagged or not tagged on certain “ports” == “associations with other stations” certainly affects whether a given frame can be multicast to multiple stations or not, or to which stations it is to be multicast.
- Similarly, this affects into which or how many queues a frame is placed, and thus is another driver of the “Multicast distribution” problem discussed above.
- It will take further discussion to resolve this.



AP-AP communications

AP-AP communications

- From the point of view of the Bridge part of a combined AP/Bridge device, there is no reason why a wireless connection to another AP/Bridge is any different from a connection to a non-AP client station of the AP.
- This implies that a data link to adjacent APs can be offered to the Bridge part of an AP.
- This slide is intended to serve as a place-holder for a discussion of this issue, if in fact there is one.



Thank you.