DRNI: Mapping between Maarten's and Steve's models

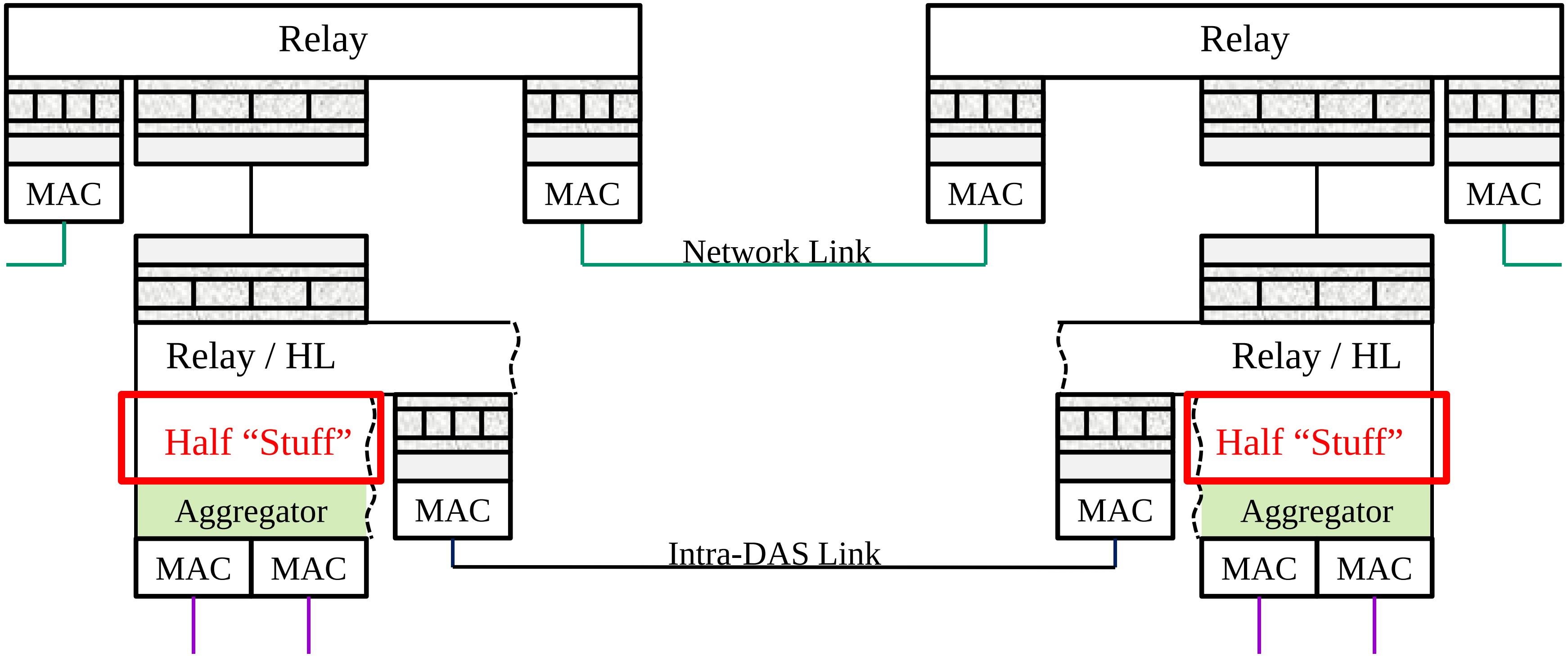
Version 01

Stephen Haddock

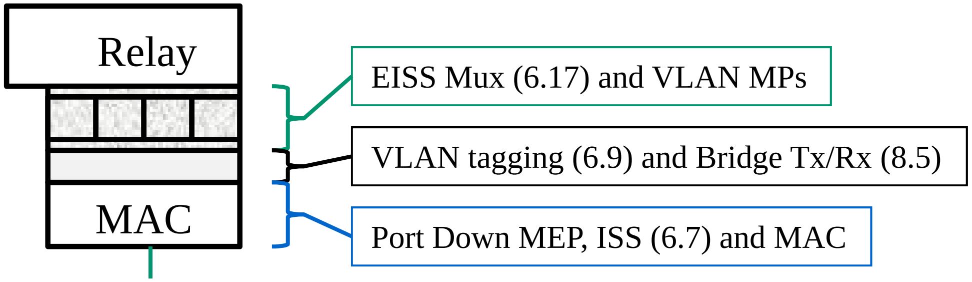
November 11, 2011

Steve’s Generalized Distributed DRNI Model

From: http://www.ieee802.org/1/files/public/docs2011/axbq-haddock-multicomponent-models-1111-v02.pdf



Key:



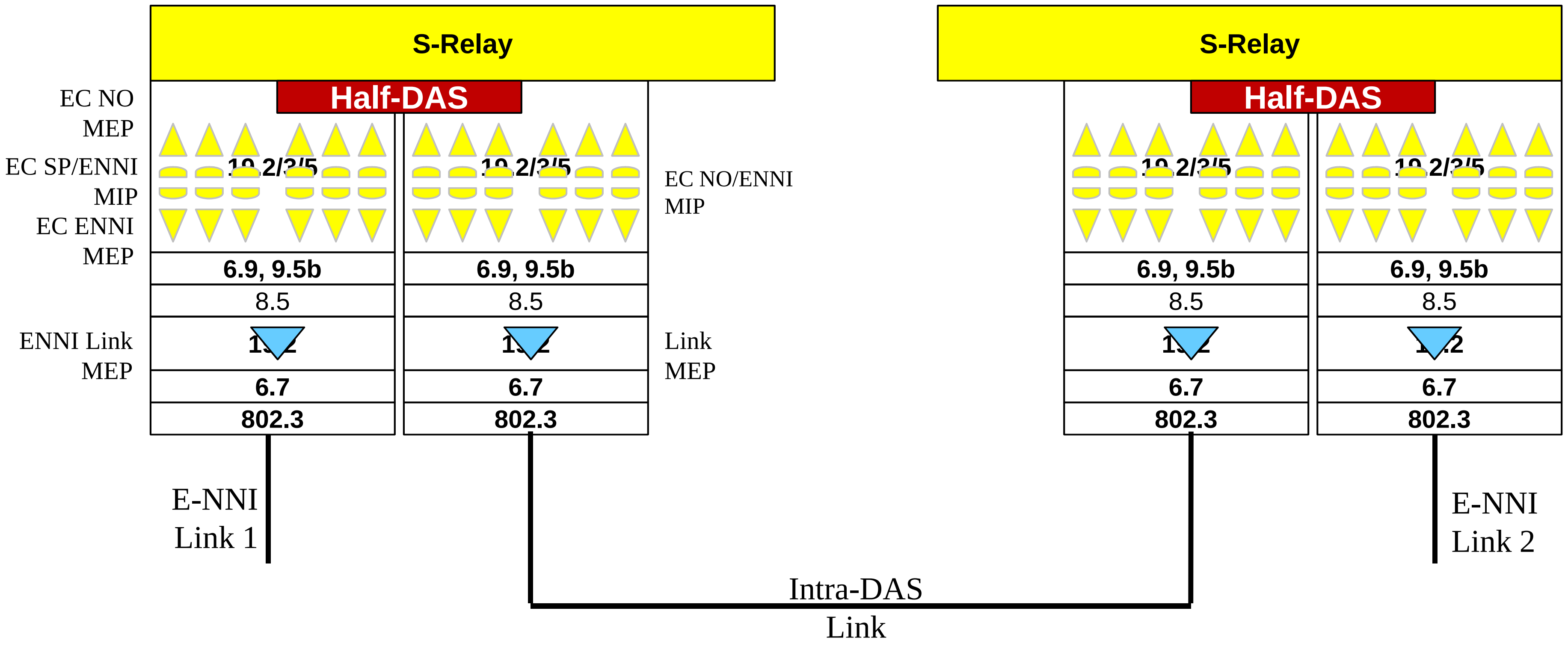
EISS Mux (6.17) and VLAN MPs

VLAN tagging (6.9) and Bridge Tx/Rx (8.5)

Port Down MEP, ISS (6.7) and MAC

Maarten’s DRNI Data Plane Model

From: http://ieee802.org/1/files/public/docs2011/axbq-vissers-drni-data-plane-model-I-and-II-comparison-1011-v00.pptx


S-Relay

Half-DAS

EC NO MEP

EC SP/ENNI MIP

EC ENNI MEP

19.2/3/5

6.9, 9.5b

8.5

ENNI Link MEP

19.2

6.7

802.3

EC NO/ENNI MIP

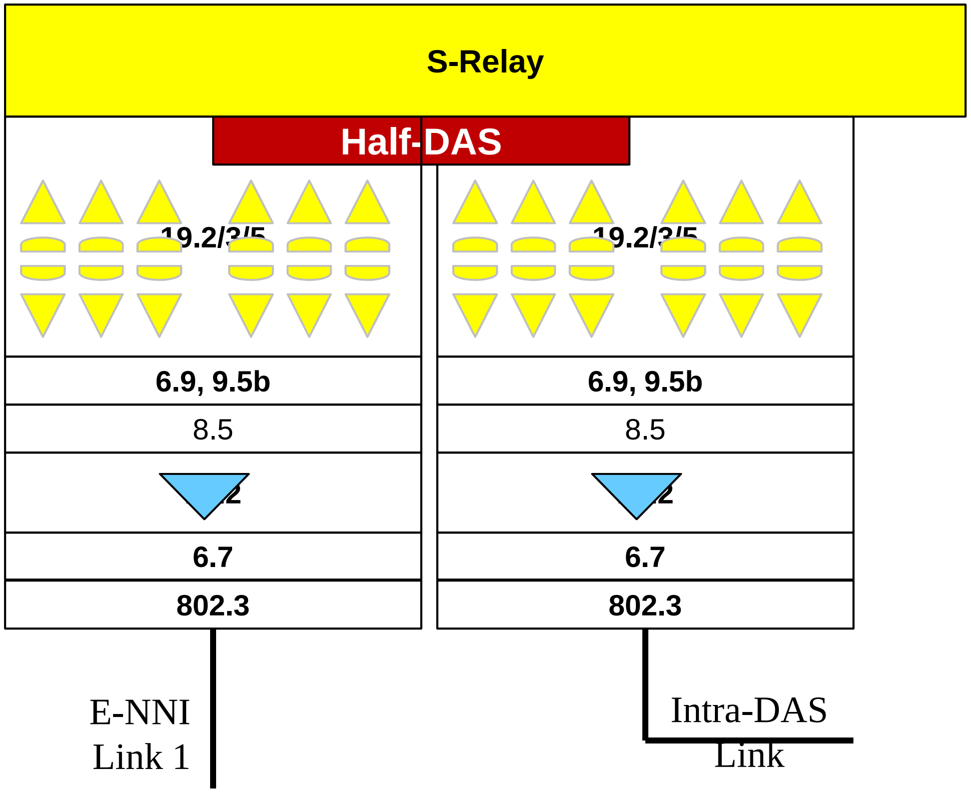
Link MEP

E-NNI Link 1

Intra-DAS Link

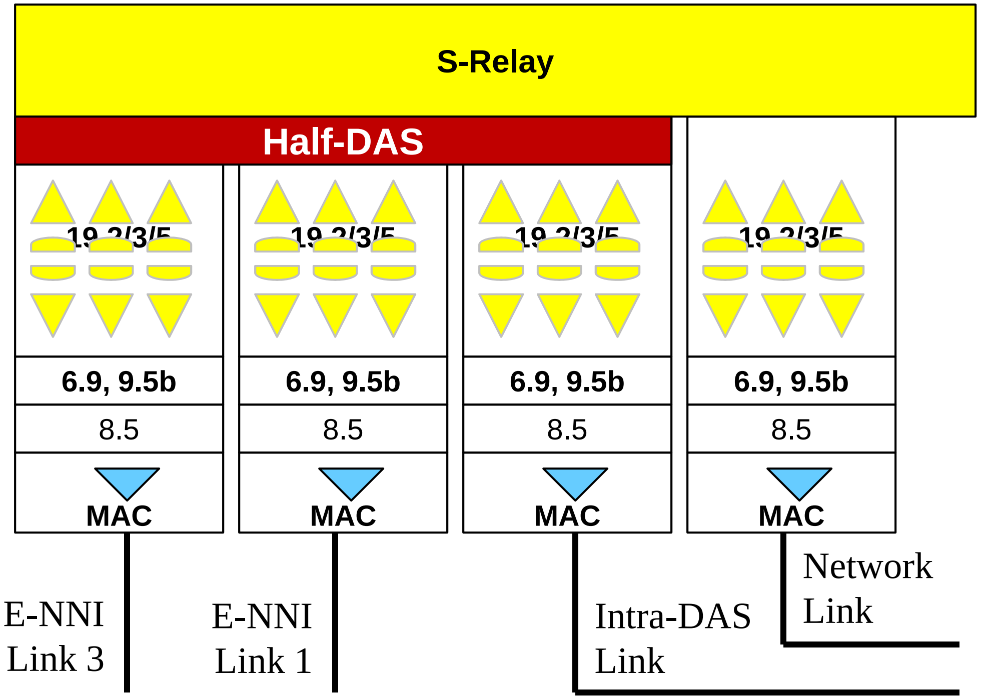
E-NNI Link 2

Reconciling the Models



Starting at the top of Maarten's model, the first question is how many Bridge Ports are on the S-Relay? I think the answer is three: one to the Half-DAS and one to portion of each of the stacks that doesn't go through the Half-DAS. The Half-DAS itself has three EISS interfaces: one up to the S-Relay and one to other portion of each of the interface stacks. This leaves two EISS interfaces at the top of each interface stack, which can be reconciled by having two instances of a 6.17 EISS Multiplex Entity above the Maintenance Point stacks: one interfacing to the Half-DAS and one interfacing directly to the S-Relay. These are merged to a single EISS below the MP stack by a single instance of the 6.17 EISS Multiplex Entity. I'll proceed assuming this interpretation.

Looking then at the left-most Bridge Port, this is the interface for unprotected services on the DRNI. This is a separate interface from the protected services that will go through the EISS to the Half-DAS. I'm not sure this is the best way to think about protected versus un-protected services. It seems like the network would consider them all to go through the same interface. Certainly two network operators would consider it one ENNI. On the other hand, if there is a MAC-in-MAC encapsulation involved , the address of the interface may change in response to some failures of the protected interface such as a "split-brain". The operators may want the unprotected services to be up or down but never change the address of the interface. If that is the case is it accurate to think of them as "on the DRNI", or do they just overlay (share?) one of the physical links of the DRNI? I propose we defer the whole unprotected service issue until we have reconciled the rest of the model.



The left side of this diagram removes the EISS for unprotected services, and adds a second ENNI link attaching to this device. Knowing that Maarten prefers to model separate physical ports with separate interface stacks to match an interface-card-and-switch-fabric style implementation, I have assumed that any additional ENNI links would be shown with separate interface stack attaching to the Half-DAS.

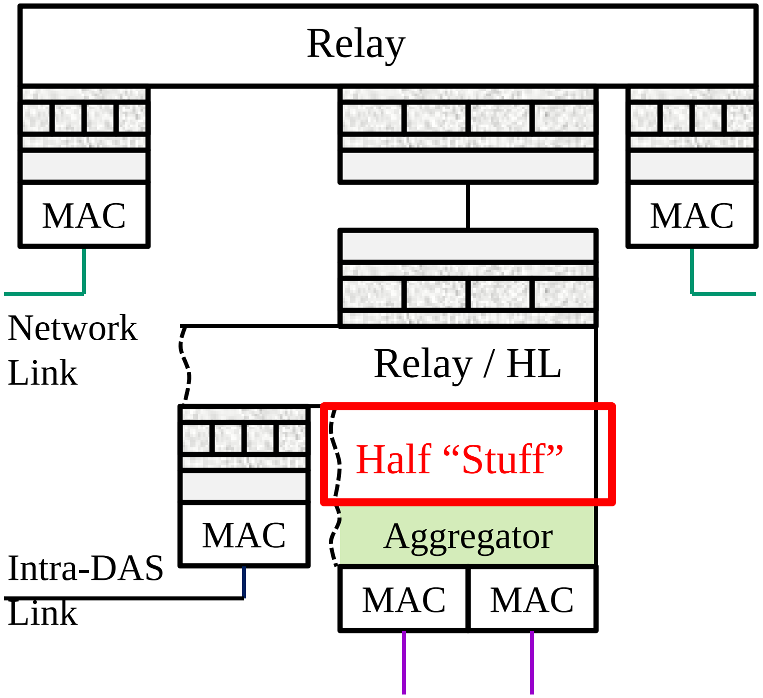
Moving to the right side of the diagram on the previous slide, the overall structure shows a single stack connecting an Intra-DAS Link at the bottom to two EISS interfaces at the top: one connecting to the Half-DAS and one directly to the S-Relay. Judging from slide 26 of Maarten's presentation, the assumption here is that the link labeled Intra-DAS Link is really shared between a virtual Intra-DAS Link and a virtual Network Link. Further it assumes that the encapsulation used to differentiate frames on the two virtual links is an S-tag, with some VIDs reserved for the Intra-DAS traffic and some VIDs reserved for network traffic. While I agree that we want to allow virtualization of the Intra-DAS Link, I don't think it appropriate for the generic model to assume virtualization, much less a specific encapsulation. Therefore I have modified the right side of the diagram on this slide to show one link as a dedicated Intra-DAS Link and another as a dedicated Network Link.

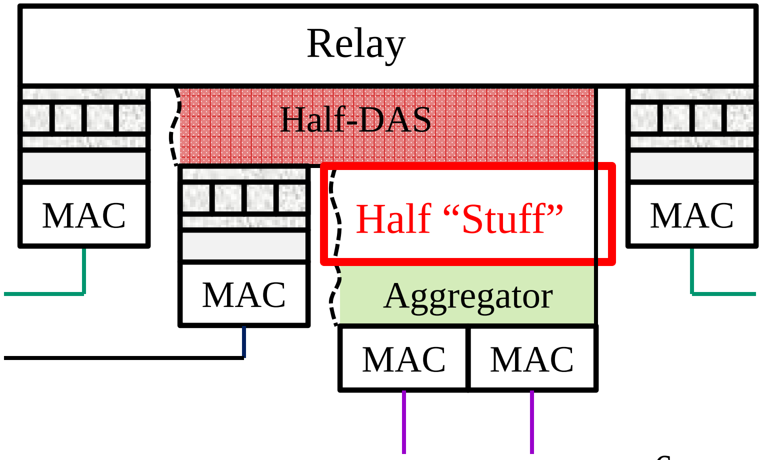
To make the model independent of link technology I have merged the 19.2, 6.7 and 802.3 boxes to a single box labeled "MAC". I have retained the blue triangle as an explicit indication that this is where Port MEPs are instantiated.

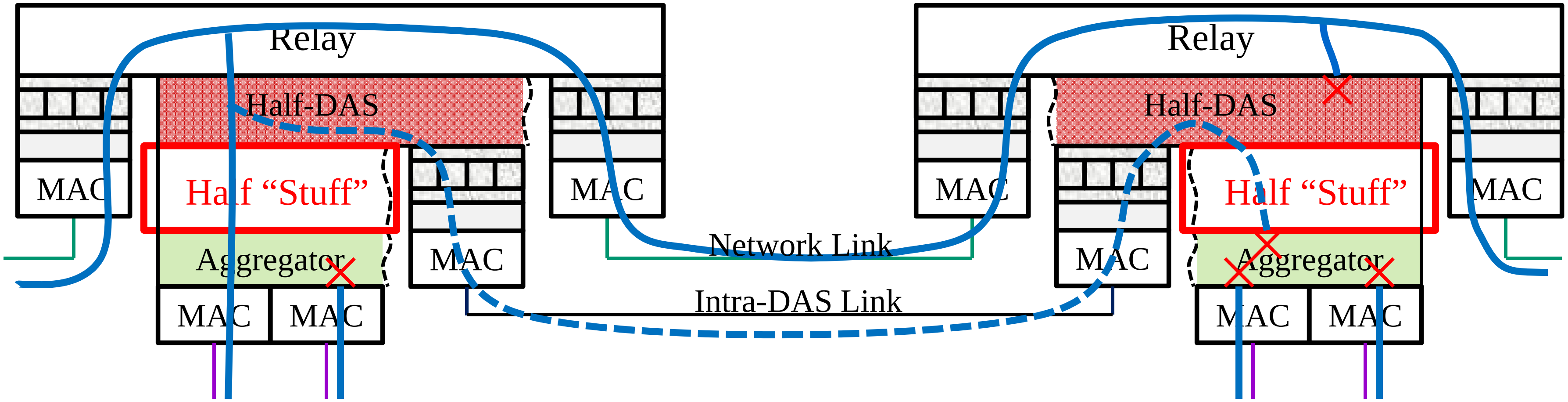
Now refer to Steve's Generalized Distributed DRNI model shown on the top to the right. The goal is going to be to replace the box labeled "Relay/HL" and all the boxes above it up to the box labeled "Relay" with a single box labeled "Half-DAS" as shown on the bottom to the right. There are two significant consequences to this seemingly simple diagrammatic change.

First, the loss of detail obscures all the entities in top diagram that become internal to the Half-DAS in the bottom diagram. For the data plane this is OK. The back-to-back VLAN-tagging (6.9) shims cancel each other out except for VID translation. If any VID translation is necessary at this point it can be specified as part of the Half-DAS functionality. The Maintenance Point stacks are only important if there is a reason to recommend instantiating MEPs or MIPs at this location, which I believe we can avoid. The MAC Relay itself will be specified as the data path functionality of the Half-DAS. More significant is obscuring the control plane, specifically the "Higher Layers" of the distributed component and the Bridge Tx/Rx (8.5.1) entities that provides Service Access Points for the Higher Layers of both the primary component and the distributed component. Although I don't think these details will be important in the final standard to define what the Half-DAS entity does, it is important to keep them in mind developing the standard to understand how and why the Half-DAS includes some control plane functionality.

Second, giving an entity in this model the same name as an entity in Maarten's model implies it performs the same functions. I think we can evolve the models so that this is true, but we are not there yet. This is evidenced by the fact that this model has an Aggregator entity above the MACs, whereas in Maarten's model the Aggregator functionality is apparently subsumed into the Half-DAS.







In this diagram the gateway functionality is completely contained in the distributed Relay (in the Half-DAS), and the link selection functionality is completely contained in the Aggregator. To do otherwise is a layer violation. This layer violation will be tolerated by allowing the Half-DAS to not send frames along a path where they would just be discarded by the Aggregator, but it will be tolerated as an optimization, not as a requirement. This is demonstrated in the above diagram by the blue lines that represent the forwarding and filtering of a broadcast frame on a particular VLAN.

Starting at the right of the diagram, a frame coming in from the network on the blue VLAN would be forwarded by the top right Relay to the Half-DAS and to the Network Link connecting the two nodes. The copy forwarded to the Half-DAS is discarded because this node is not the gateway for the blue VLAN. The other copy is received at the node on the left and forwarded to the rest of the network as well as to the left Half-DAS. This node is the gateway for the blue VLAN, so the Half-DAS (when it is not pruning) forwards a copy of the frame everywhere that it could go. In this case one copy goes down the local interface stack and one copy goes to the Intra-DAS Link. The copy going down the interface stack reaches the Aggregator and is transmitted on the link selected for that frame. The copy on the Intra-DAS Link reaches the Half-DAS in the right node. The Half-DAS determines that the frame is being transmitted on the DRNI, not received, because it was received from the node that is the gateway for the blue VLAN. Therefore it forwards the frame down the interface stack to the Aggregator. The Aggregator determines that the link selected for this frame is not attached to this node, and discards the frame. The Aggregator also discards any frames received on a DRNI Link that would not be the selected link for that frame.

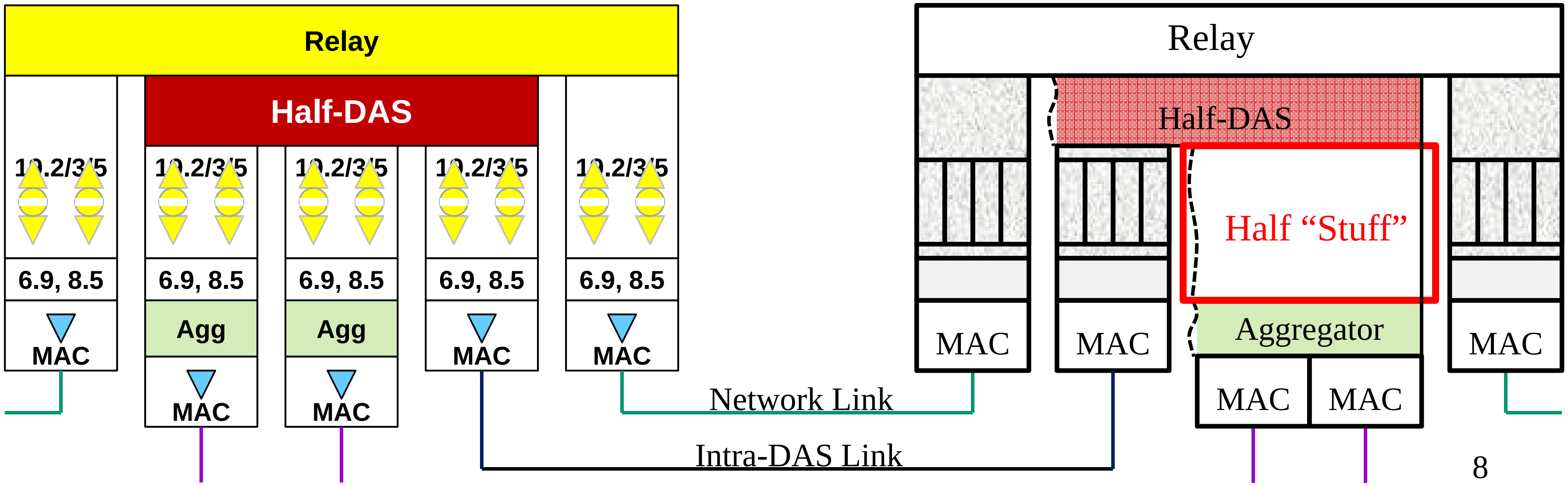
At this point the motivation for pruning is obvious. Wasting resources on the Intra-DAS Link could be avoided if the left Half-DAS could utilize the link selection criteria to not send the frame along the portion of the path shown by the dashed line. It is important that this behavior not be assumed, however, by simply embedding the Aggregator functionality in the Half-DAS.

One reason for not moving the Aggregator functionality into the Half-DAS is related to the “layer violation” aspect of pruning. Gateway selection is based on the VLAN-ID used in the network, which makes the Half-DAS the appropriate location for that functionality. In some cases the link selection is based on other service identifiers deeper in the packet. Requiring that the Half-DAS be capable of pruning based on these fields may be a burden to some implementations. Allowing the pruning but not requiring it lets each implementation make an independent decision while maintaining interoperability between two implementations making opposite decisions.

Another reason for not moving the Aggregator functionality into the Half-DAS is a control plane consideration. The Aggregator is the Service Access Point for LACPDUs. With DRNIs on multi-component bridges, ingress LACPDUs will not get to the Aggregator without being filtered, and egress LACPDUs injected at the Aggregator would be filtered or transmitted with an incorrect encapsulation. Of course an implementation could do whatever is necessary to work around this, but keeping the Aggregator just above the MAC makes the expected behavior explicit.

Based on these arguments I have taken the liberty of inserting an Aggregator shim in Maarten’s model in the diagram below. Other changes in this diagram include adding a second network link to Maarten’s diagram, and adjusting the sizes of some boxes in both diagrams to line up with each other. Steve’s model shows the generic “stuff” below the Half-DAS while Maarten’s model shows the MPs and VLAN-tagging layer appropriate for a DRNI on a single component bridge. Maarten’s model can also be generalized to just show a layer of “stuff”, but checking how well the result matches Maarten’s models for multi-component bridges has yet to be done.

The major difference between the two models in the diagram below is that Maarten’s has a separate stack for each DRNI link while Steve’s has the Aggregator multiplexing two MACs to a single interface stack. This is not a conflict. I prefer the single stack for consistency with the existing Link Aggregation specification, but the Half-DAS and Aggregator filtering and forwarding processes described on the previous slide work either way. It is completely internal to the node, and is an implementer’s choice. This is in fact the type of decision routinely made when implementing LAG.



Making the changes to each of the models described in this paper appears to completely reconcile the two models. Furthermore, considering the interface stack entities in Steve's model that were moved into the Half-DAS provides detail on the behavior of the Half-DAS. That, combined with the Half-DAS and Aggregator filtering and forwarding description, should be sufficient to begin writing text for the draft standard. (In fact the only change to the Aggregator seems to be the capability in both the Collector and Distributor to filter a frame if the link that would be selected for that frame is not attached to that Aggregator.) That said, there are still issues remaining to be resolved, some of which are listed below.

Issues

1. Is a second Intra-DAS Link at needed at the Aggregator or S-VLAN Distributed Relay to avoid having distributed MEPs?

2. Need to compare the modified models with Maarten's previous models of DRNIs on multi-component bridges.

3. Need to decide how to incorporate unprotected services into the model.

4. Need to decide whether we are going to specify how to virtualize the Intra-DAS Link, and if so what the method will be.