Multiple I-tag Registration Protocol

Multiple I-tag Registration Protocol (MIRP) for signaling among I-components

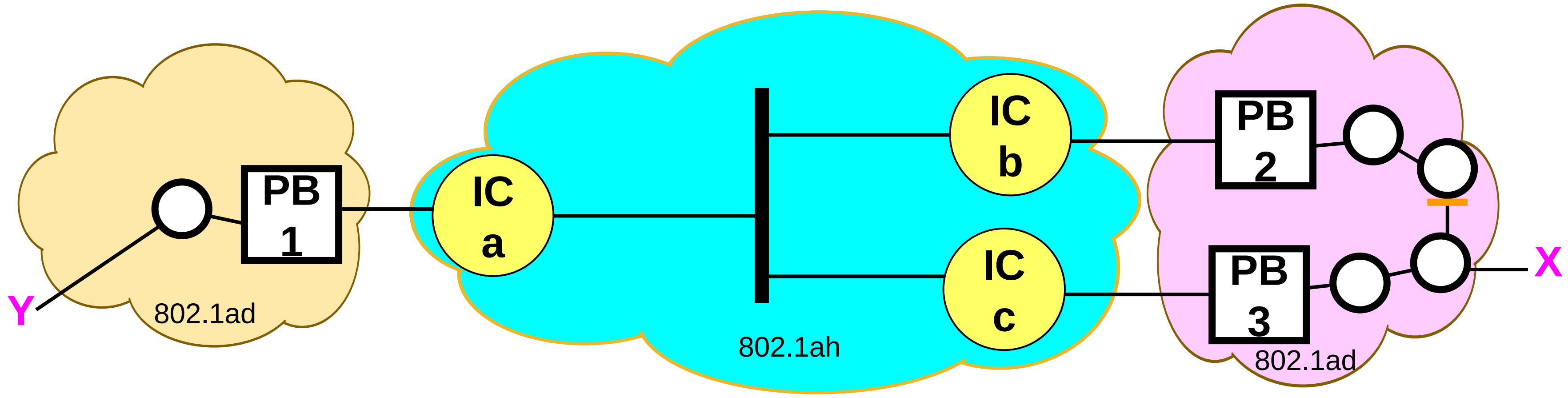
Version 1

Norman Finn

Cisco Systems

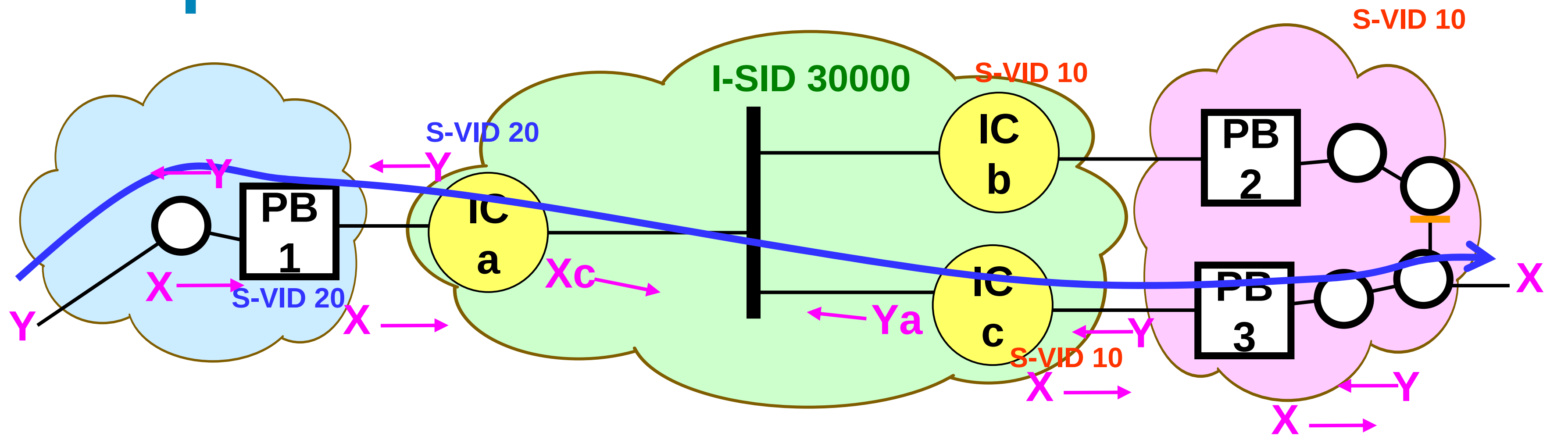
Problem statement

Simple ad-ah-ad scenario



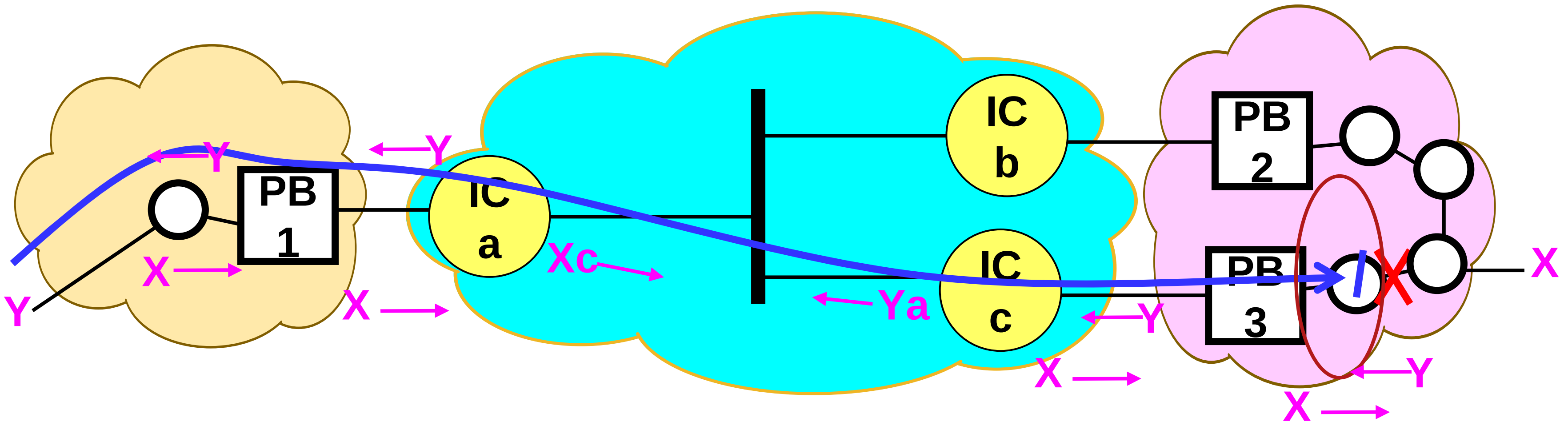
- Three I-components **a**, **b**, and **c** in an 802.1ah PBB cloud. Three S-bridges **1**, **2**, and **3** in two S-clouds. Two customer stations, **X** and **Y**.
- S-VIDs in left S-cloud are completely independent of S-VIDs in right S-cloud.
- No S-tags carried across backbone.
- Both PB **1** & PB **2** are active, each for ½ of their cloud.

Simple ad-ah-ad scenario



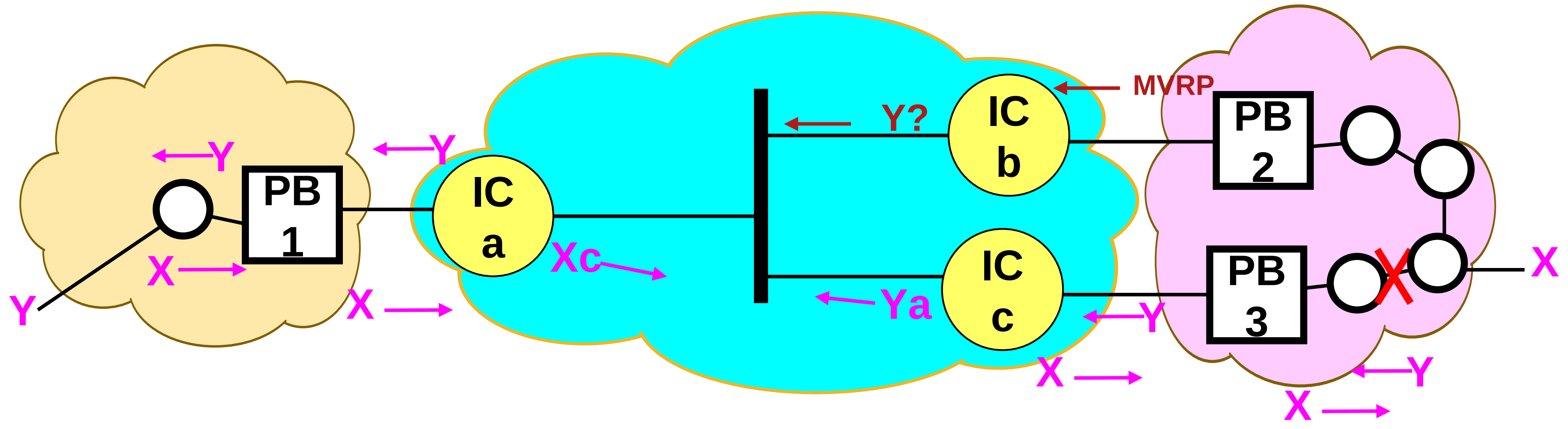
- S-VID 10 is a service in the right PB cloud.
- S-VID 20 is that same service in the left PB cloud.
- I-SID 30000 is used to carry that service across the PBB cloud.
- I-component **a** learns that X is behind I-component **c**, and I-component **c** learns that Y is behind **a**.

Simple ad-ah-ad scenario



- A change in the topology in the right cloud causes a block of MAC addresses to move from I-component **c** to I-component **b**.
- Until X sends a frame that reaches I-component **a**, all the traffic from Y to X will be blackholed, because **a** still sends it to **c**.

Simple ad-ah-ad scenario



- I-component **b** can know about the change, because PB **2** can sent it a Topology Change Notice, either via RSTP/MSTP or MVRP.
- But, how does I-component **b** notify I-components **a** and **c** that they must unlearn **X**, and start flooding?
- And, how do PB 3 and its neighbor know to unlearn **X**?

Simple ad-ah-ad scenario

- If we wait long enough, then either a station in the right cloud will speak up, and the left I-component will relearn. Often, this happens very quickly. In the worst case, however, connectivity between stations may be interrupted until the information times out.
- If this was a central S-bridge cloud connected to a pair of Customer clouds, the answer would be that the S-bridges would use MVRP to propagate a TCN for the S-VID on which the customer’s service is carried.
- The problem is that, in this case, the only service identifier that is carried among the three PB bridges is an I-SID.

Simple ad-ah-ad scenario

- In this case, I-component **b** wants to tell I-component **a**, “Forget all the MAC addresses belonging to S-VID 10 that you did not learn from me.”
- But, while the frames carrying this service are labeled with S-VID 10 in I-component **b** and I-component **c**, they are labeled with S-VID 30 in I-component **a**.
- So, if I-component **b** sends an MVRP TCN message for S-VID 10, it will be misinterpreted in I-component **a**.

Solutions

One solution

- If I-component **b** knows that its own S-VID 10 corresponds to S-VID 20 in I-component **a**, then it can translate the S-VIDs, and send an MVRP message for S-VID 20 to I-component **a**.
  - Equivalently, I-component **a** could translate from 10 to 20 on receipt of an MVRP PDU from I-component **b**.
- But, this requires every S-cloud to know the I-SID/S-VID mappings in every I-component to which it connects.
  - This information is not otherwise needed by the network.
  - It would be difficult to keep this information correct, difficult to debug when it is wrong (since it is not needed in the data plane) and difficult to reconfigure.

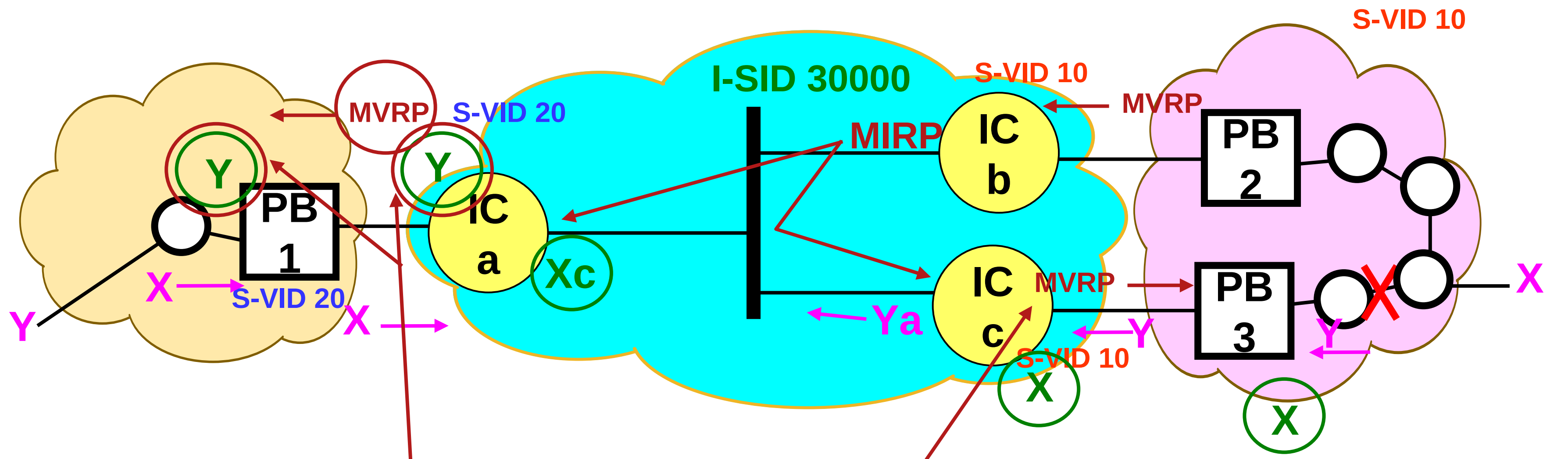
A better solution

- I-component has to know the translation from its S-VID 10 to or from I-SID 30000, since it must perform that translation on every data frame.
- It could, therefore, send an MVRP-like TCN notification for I-SID 30000 to I-components **a** and **c**.
- Similarly, I-components **a** and **c** can convert the MVRP-like TCN for I-SID 30000 back to an MVRP/STP TCN.
- This is called MIRP.

  Clearly, MIRP PDUs do not have double addresses, and do not have I-tags, because an MIRP PDU contains information about many I-SIDs.

  Clearly, MIRP PDUs are of no interest to the backbone bridges, because backbone bridges do not learn customers’ MAC addresses.

MIRP



- MIRP notifies IC **a** and IC **c** to, "Forget S-address to I-component bindings you didn't learn from me." They, in turn, notify their networks.
- Addresses are forgotten.
- Note that configuration would be required to differentiate between IC **c** and IC **a**. In this particular case, IC **c** needs to propagate the TCN into MVRP; IC **a** does not.

Simple issues

- What destination MAC address is used to get a MIRP PDU to all of the I-components that would be interested in its contents, but not necessarily all I-components?

  If information about only one I-SID is contained in the MIRP PDU, send it to the destination MAC address used for that I-SID’s multicast data frames.

  If information about more than one I-SID is contained in the MIRP PDU, send it to a newly-defined “all I-components” group MAC address.

- What B-VLAN?

  In many cases, an appropriate B-VLAN is available.

  But, it would appear that in other cases, a management, or “all I-components” B-VLAN is required.

An architectural issue

- Where does the MIRP component go in the 802.1ah architecture?
- There is no obvious place to put it.

  The I-component, itself, has one port per I-SID, and no place to send a packet that would not be encapsulated into an I-SID.

  The Virtual Instance Ports handle one I-SID each, and have no knowledge of any other I-SID.

  Clearly, the Physical Instance Port has to handle the frames, but it has no clear tie-in to the STP or MVRP state in the I-component.
- This can be resolved.

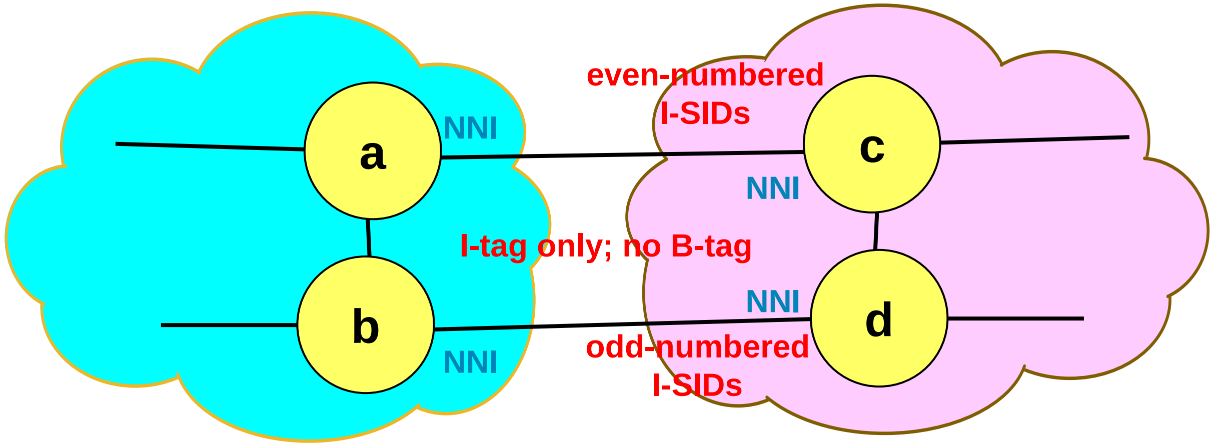
  Suggestion: Add a SAP that connects to the PIP, over which non-I-tagged traffic can pass. Since the I-component already knows about PIP IDs, this should fit the model reasonably well.

MIRP vs. MVRP

- MIRP works much like MVRP, except that:
  - It carries 24-bit I-Service IDs as its attributes, instead of 12-bit VLAN IDs.
  - As described, it is of interest only at points in the network where I-tags are added or removed, not in the interior of the 802.1ah network, so its information is propagated only by I-components.
  - In the I-components, the information is propagated by translating to/from MIRP I-SIDs and MVRP/RSTP/MSTP S-VIDs.
  - There are other uses, as well ...

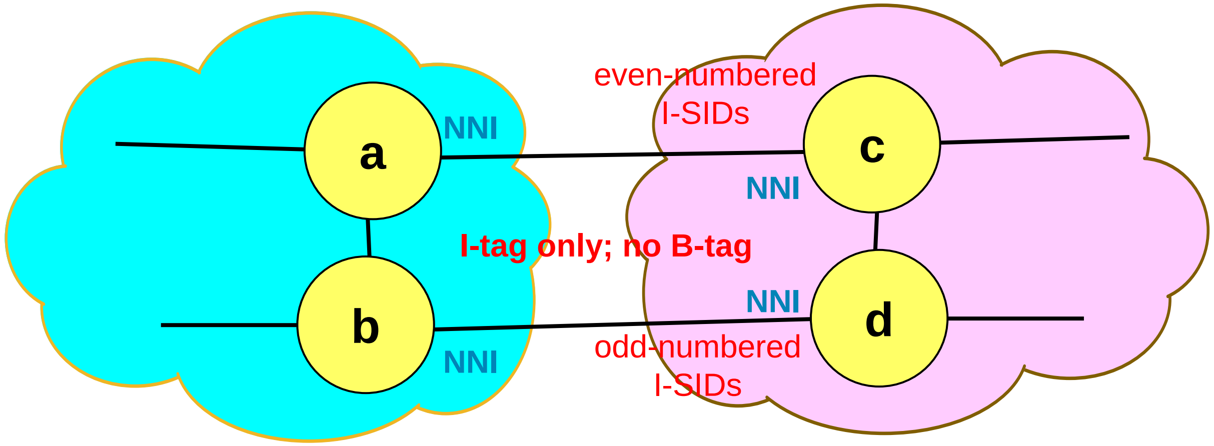
Additional uses for MIRP

A second use case: I-tagged NNI



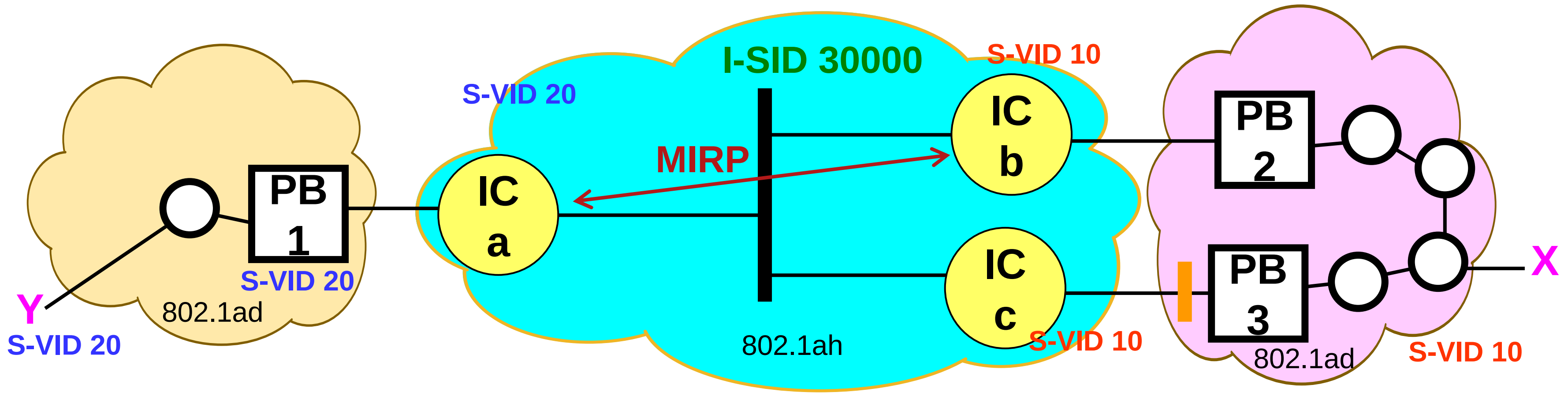
- Two backbone clouds are connected by a pair of NNIs that are I-tagged only. (Each bridge **a**-**d** supplies the B-VLAN based on looking up the I-SID in a table.)
- Bridges **c** and **d** are running L2GP (Layer 2 Gateway Protocol) to decide which B-VLANs, and hence which services, use the **a**-**c** link, and which use the **b**-**d** link.
- Bridges **a** and **b** follow the lead of Bridges **c** and **d**.

A second use case: I-tagged NNI



- How do Bridges **c** and **d** communicate their preferences to Bridges **a** and **b**, so that they don’t receive data on blocked I-SIDs, that they will just throw away?
- Answer: **MIRP**, of course.
- This is exactly the same way MVRP is used with L2GP to connect an S cloud to a Backbone cloud, or a Customer cloud to an S or Backbone cloud.

A third use case: I-component discovery



- Suppose **X** and **Y** are the only ports in a two-point service; the right-hand cloud has two connections for redundancy. PB **3** blocks one, using L2GP.
- IC **a** and IC **b** can optimize backbone efficiency by using unicast B-addresses, even for S-space multicasts.
- But, how do they discover each others' MAC addresses? Configuration? No, MIRP!

Conclusion

Conclusion

- **The problems is real.** There is currently no way to:
  - Signal the need for a remote I-component to forget associations between Customer-space MAC addresses and B-space I-component MAC addresses;
  - Relay MVRP/RSTP/MSTP TCNs from one 802.1ah Customer Network Port to another CNP in another I-component; or
  - Signal I-SID load sharing decisions across an I-tagged or I-S-tagged link.
  - Discover I-components belonging to the same point-to-point services across the backbone.
- **MIRP would solve these problems.**