Bridges and End-to-End OAM

Norman Finn, Cisco Systems

End-to-End OAM

- **Existing (but little-known) features of IEEE Std. 802.2-1998.**
- **Current IEEE 802.3ah OAM draft 1.3**
- **MEF working documents not referenced, here, but they are a good place to start. See "04067_000_Ethernet Service OAM_L6_Wils.doc", in particular.**

IEEE Std. 802.2-1998

- **Provides "XID" (eXchange IDentification) and "TEST" commands, which *are* end-to end!**
- **XID is a basic Ping. May be sent to an individual MAC address, to a multicast, or to the broadcast address.**
- **XID includes coded information about the LLC capabilities such as LLC2 capability and buffer size.**
- **TEST is similar, and allows an arbitrary amount (up to the max frame size) of data to be reflected.**
- **In both commands, the receiver swaps MAC addresses and SAPs, sets "response" bit, and replies to, rather than simply echoing, the request.**

IEEE 802.3ah OAM draft 1.3

- **Sits immediately above MAC Control**
- **Uses a MAC address that cannot pass through a bridge**
- **OAM Discovery method**
  - **Simple exchange of capabilities.**
  - **One important capability: Can I send OAM (e.g. "dying gasp" event notification) when the receive link indicates a failure? (Some chip sets disable the link when it is unidirectional.)**
- **OAM Loopback**
  - **One device (Active) sets other device into Loopback Mode**
  - **Device in Loopback Mode passes no frame down (out) its stack from upper layers. It reflects all frames received from wire back onto wire *verbatim* (MAC addresses not swapped).**

IEEE 802.3ah OAM draft 1.3 (contd.)

- **Variable Request/Response**
  - **Allows one end to read, but not set, MIB variables in the other end.**
- **Events**
  - **Critical events: Link fault, dying gasp.**
  - **Non-critical events: Error condition thresholds exceeded.**
- **Given device may be configured to be Active or Passive**
  - **Passive device not allowed to initiate Discovery procedure, inquire about variables, or set loopback mode.**
  - **Clearly, one expects CE to be Passive, and PE to be Active.**
- **Vendor-specific commands**
  - **Catch-all for adding absolutely anything you want to add.**

What We Want to Learn from E2E OAM

- **"Ping" type questions:**
  - **Is Customer device (CE) {MAC address, VLAN} {X,Y} reachable from Customer device {Z,Y}?**
  - **Is CE {X,Y} reachable from Provider node (PE) Z?**
  - **Is the PE closest to CE {X,Y} reachable from PE Z?**
  - **What is the PE immediately adjacent to CE {X,Y}?**
  - **Is PE X reachable via the Customer's data path from PE Y?**
  - **What is (are) the (other) PE(s) of this Customer Service Instance which have UNIs on this Instance?**
- **"Traceroute" type questions:**
  - **For each of the above "Ping" type questions, what intermediate PEs handle customer data along the path(s) of the Pings?**
  - **If a "Ping" question's answer is, "No," where does the Ping fail?**

What We Want to Learn from E2E OAM

- **"Quality of Service" or "Service Level Agreement" type questions:**
  - **For each of the above "Ping" type questions, at what data rate are the customer's data frames being carried? Per QoS level?**
  - **With what probability of frame loss? Per QoS level?**
  - **With what bit error rate? Per QoS level?**
  - **With what delay? Per QoS level?**
  - **With what variation of delay (jitter)? Per QoS level?**
- **"Alarm" type notifications**
  - **Tell some number of other entities, who may be interested, that I am (not) having certain problems.**

Types of End-to-End OAM PDUs

- **In-Band OAMPDUs:** **Look more or less like Customer data frames. May be distinguished from Customer data frames by Ethertype and/or Destination MAC address.**
- **Out-of-Band OAMPDUs:** **Are distinguished from Customer Data frames by some characteristic not lying within the bounds of a Customer data frame [Destination MAC address through Frame Check Sequence].**

Comparing E2E OAMPDU Types

- **In-Band OAMPDUs:**
  - **Can work end-to-end across Provider network(s) utilizing different technologies, e.g. both Q-in-Q and EEoMPLS.**
  - **May be hard for Provider's nodes to distinguish them from Customer data frames, and/or prevent from egressing the PN.**
  - **May make use of more of the normal Customer data frame forwarding mechanism, and thus stay closer to the path taken by actual Customer data frames.**
- **Out-of-Band OAMPDUs:**
  - **Can work end-to-end only if a single technology is utilized in the Provider network(s).**
  - **Are easily distinguished from Customer data frames, and easily prevented from egressing the Provider network(s).**
  - **May not pass through the normal Customer data frame path.**

Examples of E2E OAMPDU Types

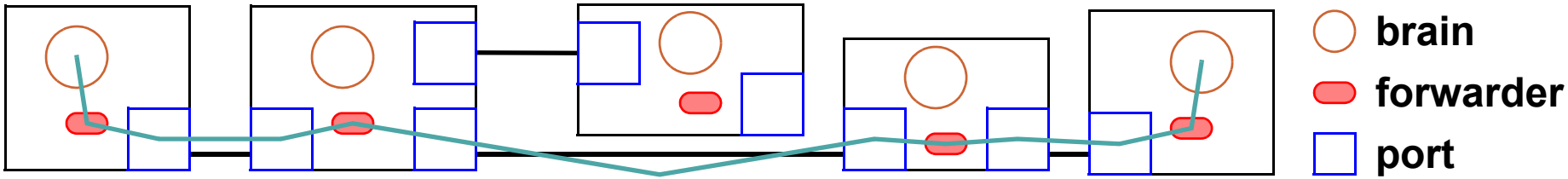
- **In-Band OAMPDUs:**
  - **IEEE 802.2 XID and TEST frames. (End-to-end)**
  - **IEEE 802.3ah Draft 1.3 OAM PDUs. (Link local)**
  - **IP Pings. (End-to-end, but at Layer 3)**
- **Out-of-Band OAMPDUs:**
  - **ATM OAM Cells.**
  - **MPLS Control packets.**
  - **Cisco's CDL management channel.**

End-to-End OAMPDU Relay Models

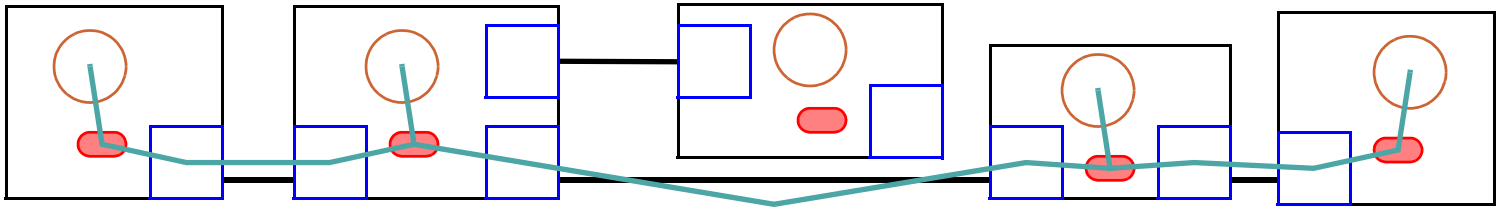
- **Simple OAM:** OAMPDU is addressed to one or more Provider or Customer devices, which answer it.
- **Spied-upon OAM:** As the OAMPDU travels through the Provider network, each node's "brain" receives a copy of it, while the OAMPDU is forwarded through the normal path for Customer data frames.
- **Relayed OAM:** As the OAMPDU travels through the Provider network, each node stops it and examines it. Each node may answer it, modify it, and/or send it to one or more other nodes towards its destination.
- An OAMPDU may or may not be allowed to egress the Provider's network and reach the Customer.

Illustrating E2E OAM Relay Models

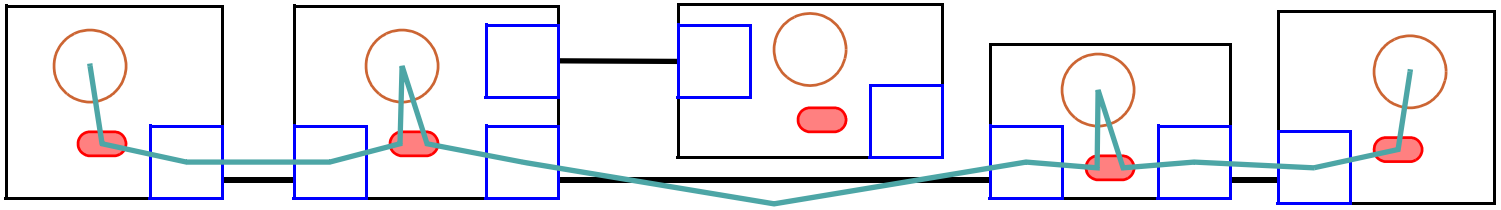
- **Simple OAM**



- **Spied-upon OAM**

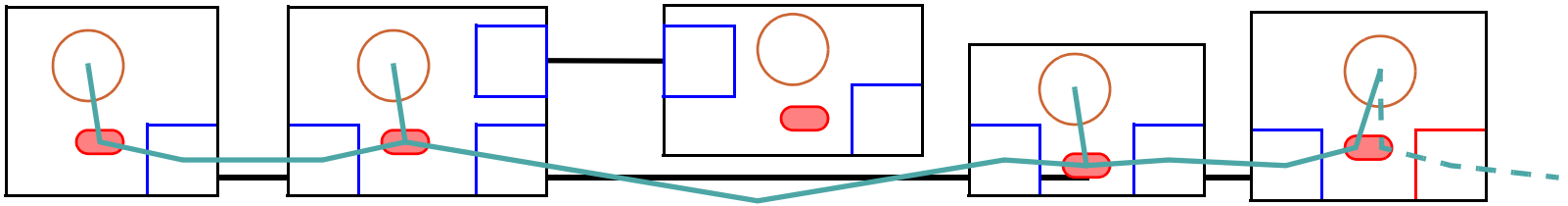


- **Relayed OAM**

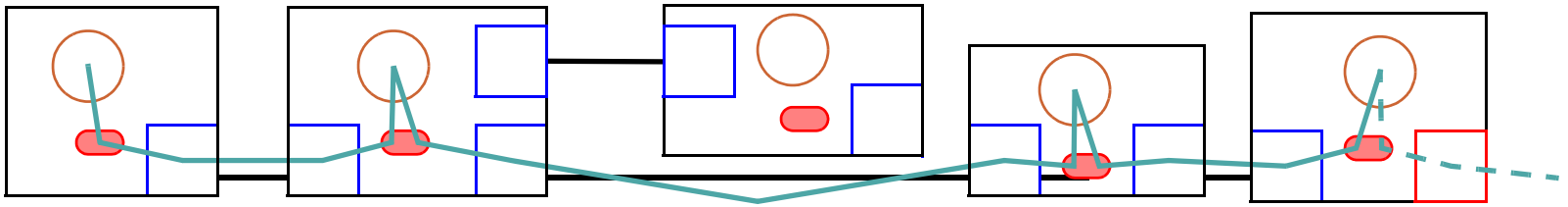


Combination E2E OAM Relay Model

- **Intercept-at-end OAM is equivalent either to Spied-upon OAM with a filter to prevent egress, or to Relayed OAM, where only the egress PE performs the Relay function:**



- **or:**



Examples of E2E OAMPDU Relay Models

- **Simple OAM**
  - **XID or TEST sent to a unicast address.**
- **Spied-upon OAM**
  - **XID or TEST sent to an appropriate multicast address.**
- **Relayed OAM**
  - **Bridge Protocol Data Units (BPDUs).**
- **Note that Link Local OAMPDUs are "none of the above"**

Capabilities of E2E OAM Relay Models

- **Simple OAM:** Cannot use Customer MAC addresses (except to contact Customer Equipment), so cannot follow exactly the same path as Customer data frames.
- **Spied-upon OAM:** Follows the data path of Customer data frames most closely, and therefore most desirable model.
- **Relayed OAM:** Cannot use Customer MAC addresses, so cannot follow exactly the same path as Customer data frames. Heavyweight, as each intermediate PE must process the OAMPDU.

Implementation of E2E OAM Relay Models

- **Simple OAM: Within the current capabilities of existing devices, because destination MAC address of the "Ping" targets the device(s) that should answer it.**
- **Spied-upon OAM: May or may not be within the current capabilities of existing devices.**
  - **If Ping uses a multicast destination MAC address, intermediate PEs should be able to take a copy.**
  - **If Ping uses Customer's MAC addresses and/or VLANs, intermediate PE may or may not be able to detect EtherType.**
- **Relayed OAM: Within the current capabilities of existing devices, just as BPDUs are.**

Traceroute

- **IP Traceroute is an application which utilizes the IP Time-To-Live (TTL) field in a Ping.**
  - **Not easily applicable to IEEE 802, as we have no TTL.**
  - **Could be adapted to a Relayed OAM scheme.**
- **Current "Layer 2 Traceroutes" are actually management functions which use SNMP to explore the forwarding tables of each bridge along the path.**
  - **Requires knowledge of network topology, presumably obtained from LLDP.**
- **New Layer 2 Traceroute could be Spied-upon OAM with all devices responding with own name and name of next hop, using LLDP information.**

Take a Lesson from Token Ring?

- **Perhaps a better Traceroute function would be to generate a packet, distinguished by Ethertype, and addressed to a Customer MAC address, either unicast or Multicast.**
- **This frame is modified by the port or forwarding hardware of each Provider Bridge to identify the bridge, exit, and/or entrance port it passed through.**
- **When this frame either reaches its destination or would be discarded, it is reflected back to the source.**
- **Problems:**
  - **What if the frame is lost by something which cannot reflect it?**
  - **This is a more significant hardware change than other schemes.**

Judgement Call

- **Very Best: Spied-upon OAMPDU using Customer's unicast or non-unicast destination MAC address.**
  - **Bad news: This requires that intermediate nodes recognize EtherType buried one or two .1Q-like tags deep, which may require a hardware change to the typical Provider Bridge.**
- **Second Best: Spied-upon OAMPDU using Provider's multicast MAC address, carrying Customer's MAC address as payload.**
  - **Bad news: The intermediate nodes' "brains" report what they *think* should happen to the Customer's packet, which may not be what *really* happens to it.**