Logical MACs, LLIDs and EPONs

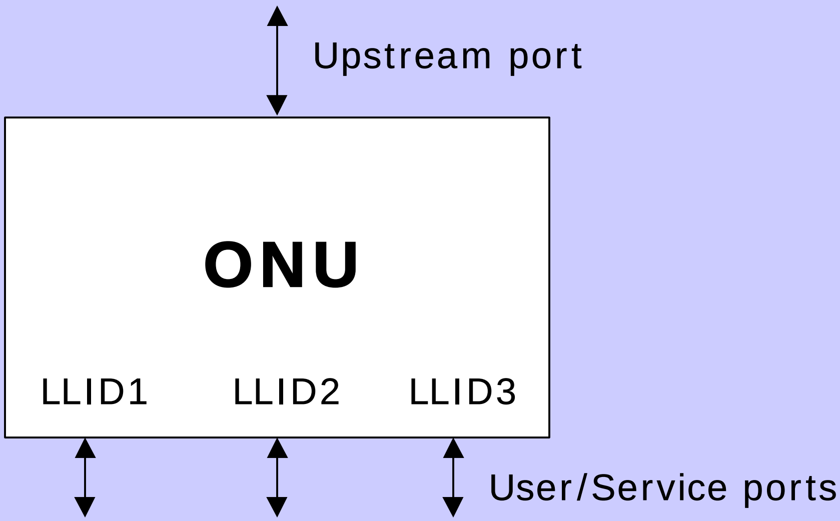
Bob Gaglianello

Why Logical MACs (a brief History)

- **LMACs were introduced for 802.1D Bridge compliance**
  - Both P2PE and efficient SE can now be supported with “simple” filtering rules (Sala_2_0502.pdf)
- **Placed in preamble in either direction within the PON**
  - LLID is term currently used for tag in preamble.
- **Since then, concept of multiple LLID’s within a single ONU has been introduced to:**
  - Resolve Open Access requirements
  - Provide Privacy through separation of user data connected to same ONU.
  - Provide a mechanism to implement SLA’s, QOS, etc.

ONU with Multiple Logical/Physical ports

- **The Resulting ONU Looks like:**



- **How does OLT Address and Schedule Multiple LLID's within a single ONU ?**
  - Several methods have been proposed

First Method

- **OLT Treats Each LLID as Separate / Independent ONU:**
  - Use MPCP discovery, gate/report, for each LLID
    - LLID carried in preamble
      - Not visible to higher layers since preamble is remove
  - This is an elegant and simple solution for scheduling and addressing
    - Unfortunately, it has scalability problems due to guard bands required between each MAC's upstream data transmission
      - described / quantified on next slide
    - Possible compliance problems with 802.1D bridging
      - An ONU with Multiple "user" ports and a single upstream port is exactly the same topology as EPON, which caused compliance problems with 802.1D bridging

Number of Schedulable Entities Impact on Link Efficiency

- **Link efficiency is impacted by number schedulable entities in a PON due to the need for Guard-Bands separating traffic from different entities**
  - **Assume:**
    - **1 msec “Gate-Report Cycles”**
    - **1 usec Guard-Band**
    - **64 ONU’s**
    - **1, 8, or 16 LMACs per ONU (64, 512, 0r 1024 schedulable entities)**
  - **Total time taken up by Guard-Bands is:**
    - 1000 * 1e-6 * 64 * 1 = 64 msecs ==> 6.4 % overhead
    - 1000 * 1e-6 * 64 * 8 = 512 msecs ==> 51.2 % overhead
    - 1000 * 1e-6 * 64 * 16 = 1024 msecs ==> 102.4 % overhead
  - Obviously, the number of schedulable entities needs to be small to obtain reasonable link efficiencies.

Second Method

- **OLT Schedules all LLID's from a given ONU as a contiguous burst**
  - This relieves inefficiency / scalability problems
    - Gate / Report formats need to be updated
    - Added complexity to scheduling algorithm
  - LLID's still carried in preamble
    - Still not visible to higher layers since preamble is removed
  - Still Possible compliance problems with 802.1D bridging
      - An ONU with Multiple "user" ports and a single upstream port is exactly the same topology as EPON, which caused compliance problems with 802.1D bridging

Third Method

- **OLT Schedule's ONU as a single entity**
  - This relieves inefficiency / scalability problems
    - Added complexity to scheduling algorithm at ONU
    - Gate / Report formats simplified because of single entity
  - To implement multiple physical ports, one must use 802.1D bridge
- **Use VLAN or VLAN-like techniques to provide open access, Privacy, and implement SLA's, QOS, etc.**
  - See (Bemmel_1_0502.pdf) and (Kim_1_0502.pdf)
  - 802.1Q VLANs can address these issues
    - VLANs are visible to L2 and provide an interface to higher layers
    - VLAN-based traffic segregation, prioritization and rate limiting techniques are available
    - VLAN limitations can be addressed in 802.1Q

Summary

- **Three methods were briefly described to provide open access, Privacy, and implement SLA's, QOS, etc.**
- **Base on link efficiency calculations, the argument for a single LLID per ONU coupled with VLAN techniques seems to provide the best solution for EPON.**