Reassembly Buffer and Working Mechanism

Contributors:
Wei Liang Zhang, ZTE
Jun Shan Wey, ZTE
Li Quan Yuan, ZTE
Yong Guo, ZTE

Supporter:
Frank Effenberger, Huawei
Duane Remein, Huawei

Background

- kramer_3ca_1_0117_sar_01.pdf discusses dimensioning of reassembly buffers at the OLT, points out the reassembly buffer problem and presents several methods.
  - How to support a large number of ULIDs/PLIDs while keeping memory requirements reasonable?
  - What is the reasonable reassembly buffer size?
- Other ideas are called for.

Problem Statement

- 100G-EPON supports up to 56K ULIDs and 4K PLIDs (see kramer_3ca_1b_0916.pdf)
- 100G-EPON needs to support jumbo frames (10KBytes)
- Static memory allocation is not feasible
  - 60K x 10KB = 600 MBytes
- **How to support a large number of ULIDs/PLIDs while keeping memory requirements reasonable?**
- **What is the reasonable reassembly buffer size?**
  - 2MB, 4MB?
  - Examples in this presentation will assume 4MB reassembly buffer per PON port

Summary

- Method #1 – Static Flow Limit
- Method #1a – Static Flow+Frame Limit
- Method #2 – Dynamic Slot Reservation
- Method #2a – Dynamic Allocation
- Method #3 – Static Frame/Dynamic Slot
- Method #4 – Token Granting

- Any other ideas?

- Straw Polls

Concept of fragmentable flow

- kramer_3ca_1_0117_sar_01 introduces the concept of fragmentable flow.

- A fragmentable flow
  - may generate fragments on ONU side and need reassembly buffer on OLT side.
  - means a ULID which is allowed to send fragments.

- The number of fragmentable flows
  - equals the number of ULIDs supporting fragment.
  - decides the size of needed reassembly buffer.
  - should not overload the maximum reassembly buffer.

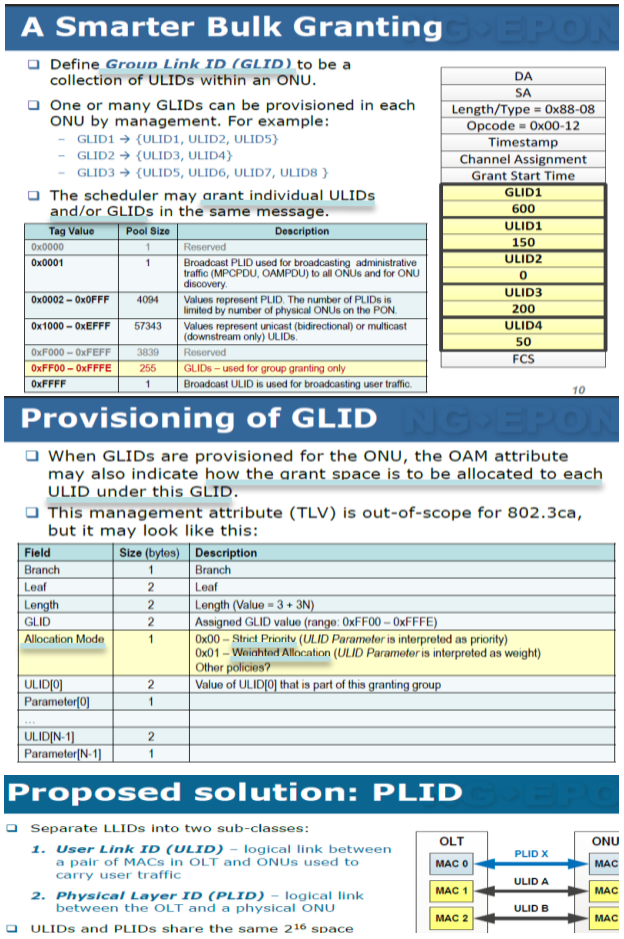
- How to support a large number of ULIDs/PLIDs while keeping memory requirements reasonable?
  - To limit the number of fragmentable flows. HOW?

Method #1 – Static Flow Limit

- Limit the number of fragmentable flows
- Add a field to the ULID assignment attribute to indicate whether that ULID is allowed to send fragments or not.
- Each fragmentable ULID gets a 10K reassembly slot permanently assigned to it.
  - With 4MB buffer and 10KB max frame size, we can have 400 fragmentable ULIDs per PON.
- 400 ULIDs out of 56K is just 0.7%.
- Unfragmentable ULIDs constitute the majority and will cause transmission inefficiency because OLT does not know the frame boundaries
- Hooks required from 802.3ca: a flag (boolean field) in the ULID assignment attribute

Concept of Group Link ID (GLID)

- kramer_3ca_1a_0916_PLID_ULID shows the concept of Group Link ID (GLID).
  - A GLID is a collection of ULIDs within an ONU.
  - OLT may grant individual ULIDs and/or GLIDs.
  - The grant for a GLID is further allocated to member ULIDs according to the configured allocation mode.
  - A member ULID still represents an individual fragmentable flow.
- Is it possible that member ULIDs are not granted?
  - Member ULIDs share the grant for their GLID.
  - A member ULID does not represent an individual fragmentable flow any more.
- Note: PLIDs and ULIDs are both LLIDs, so a PLID could be independent or merged into a GLID.



A Smarter Bulk Granting

- Define *Group Link ID (GLID)* to be a collection of ULIDs within an ONU.
- One or many GLIDs can be provisioned in each ONU by management. For example:
  - GLID1 → {ULID1, ULID2, ULID5}
  - GLID2 → {ULID3, ULID4}
  - GLID3 → {ULID5, ULID6, ULID7, ULID8 }
- The scheduler may grant individual ULIDs and/or GLIDs in the same message.

Tag Value	Pool Size	Description
0x0000	1	Reserved
0x0001	1	Broadcast PLID used for broadcasting administrative traffic (MPCPDU, OAMPDU) to all ONUs and for ONU discovery.
0x0002 – 0x0FFF	4094	Values represent PLID. The number of PLIDs is limited by number of physical ONUs on the PON.
0x1000 – 0xEFFF	57343	Values represent unicast (bidirectional) or multicast (downstream only) ULIDs.
0xF000 – 0xFEFF	3839	Reserved
0xFF00 – 0xFFFE	255	GLIDs – used for group granting only
0xFFFF	1	Broadcast ULID is used for broadcasting user traffic.

DA
SA
Length/Type = 0x88-08
Opcode = 0x00-12
Timestamp
Channel Assignment
Grant Start Time
GLID1
600
ULID1
150
ULID2
0
ULID3
200
ULID4
50
FCS

10

Provisioning of GLID

- When GLIDs are provisioned for the ONU, the OAM attribute may also indicate how the grant space is to be allocated to each ULID under this GLID.
- This management attribute (TLV) is out-of-scope for 802.3ca, but it may look like this:

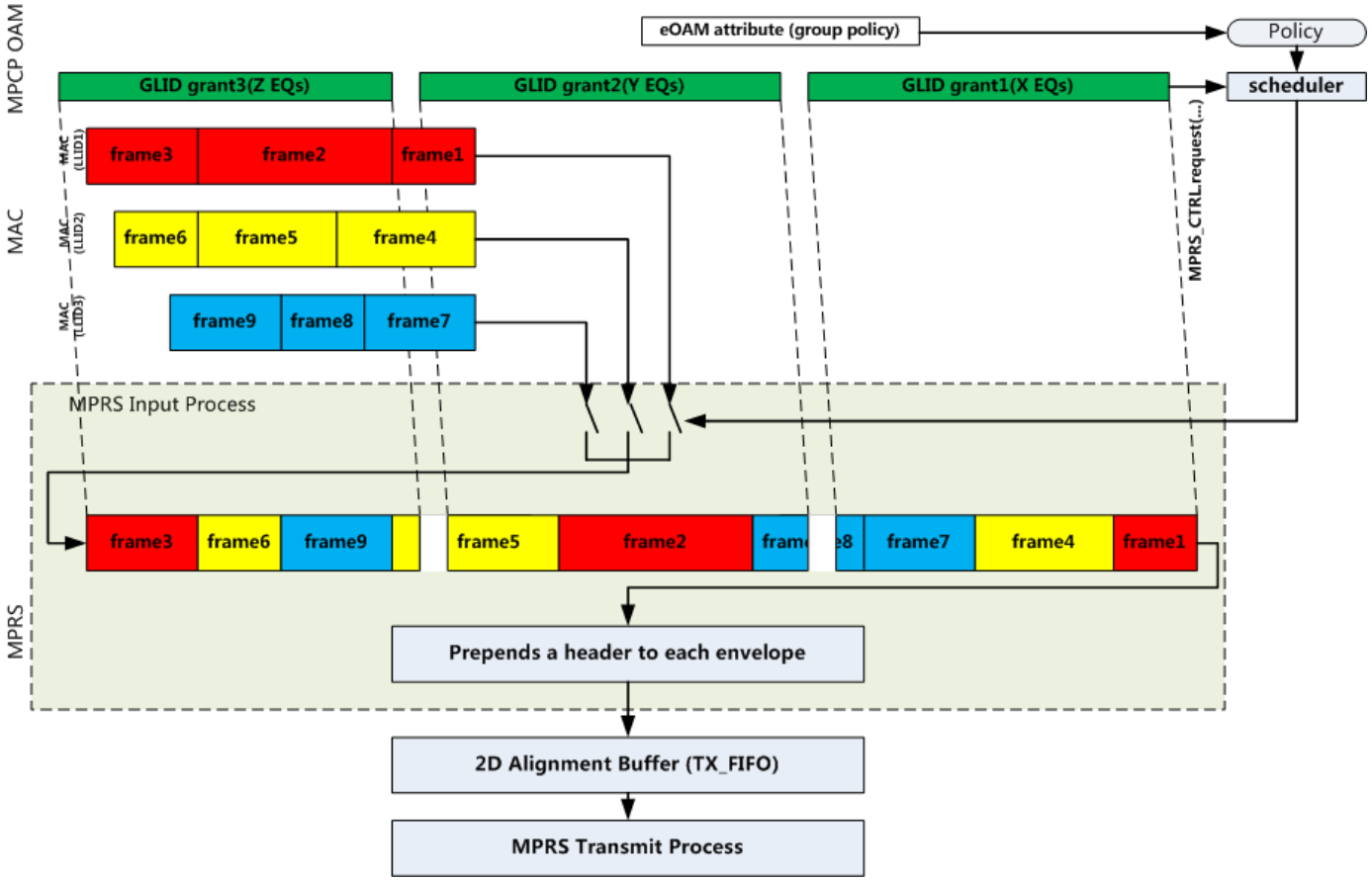
Field	Size (bytes)	Description
Branch	1	Branch
Leaf	2	Leaf
Length	2	Length (Value = 3 + 3N)
GLID	2	Assigned GLID value (range: 0xFF00 – 0xFFFE)
Allocation Mode	1	0x00 – Strict Priority (*ULID Parameter* is interpreted as priority) 0x01 – Weighted Allocation (*ULID Parameter* is interpreted as weight) Other policies?
ULID[0]	2	Value of ULID[0] that is part of this granting group
Parameter[0]	1	
…		
ULID[N-1]	2	
Parameter[N-1]	1	

Proposed solution: PLID

- Separate LLIDs into two sub-classes:
  1. ***User Link ID (ULID)*** – logical link between a pair of MACs in OLT and ONUs used to carry user traffic
  2. ***Physical Layer ID (PLID)*** – logical link between the OLT and a physical ONU
- ULIDs and PLIDs share the same 2^{16} space

Member ULIDs of one GLID share grants

- Member ULIDs of one GLID share grants.
  - MAC frames from member ULIDs are scheduled into a queue of GLID MAC frames, based on scheduling policies (e.g., strict priority or weight of each member ULID).
  - ONU sends frames and/or fragments from GLID MAC frame queue to OLT.
  - GLID is granted but its member ULIDs share the grant.



Member ULIDs of one GLID share grants (continued)

- Fragment is potentially generated at the end of a GLID grant, and the remaining fragment is sent at the beginning of the next GLID grant.
  - Member ULIDs potentially support fragment.
  - Any member ULID does not represent an individual fragmentable flow.
  - A GLID represents an individual fragmentable flow which needs reassembly buffer on OLT side.
- Only independent ULIDs and GLIDs represent individual fragmentable flows.
- By grouping ULIDs into a GLID, the number of fragmentable flows can be reduced.
- At minimal, the number of fragmentable flows equals the number of ONUs, because an ONU supports at least one fragmentable flow.

Reasonable reassembly buffer size / fragmentable flows

- The reassembly buffer size is designed by OLT vendors, based on operators' requirements.
  - Operators are encouraged to provide the requirements of reassembly buffer.
- When the configured ULIDs do not overload the reassembly buffer, they are individually granted, otherwise, some of them should be grouped into one or more GLID(s), based on operators' policy.
- The reassembly buffer should at least support one fragmentable flow per ONU.

Summary and proposals

- The size of reassembly buffer is designed by OLT vendors. We propose

1. The reassembly buffer should support at least one fragmentable flow per ONU.

2. Multiple ULIDs can be grouped into one GLID, to reduce the number of fragmentable flows.
   - OLT grants GLIDs instead of its member ULIDs.
   - MAC frames of member ULIDs are scheduled into one queue of MAC frames, to share the GLID grant.

Thank You