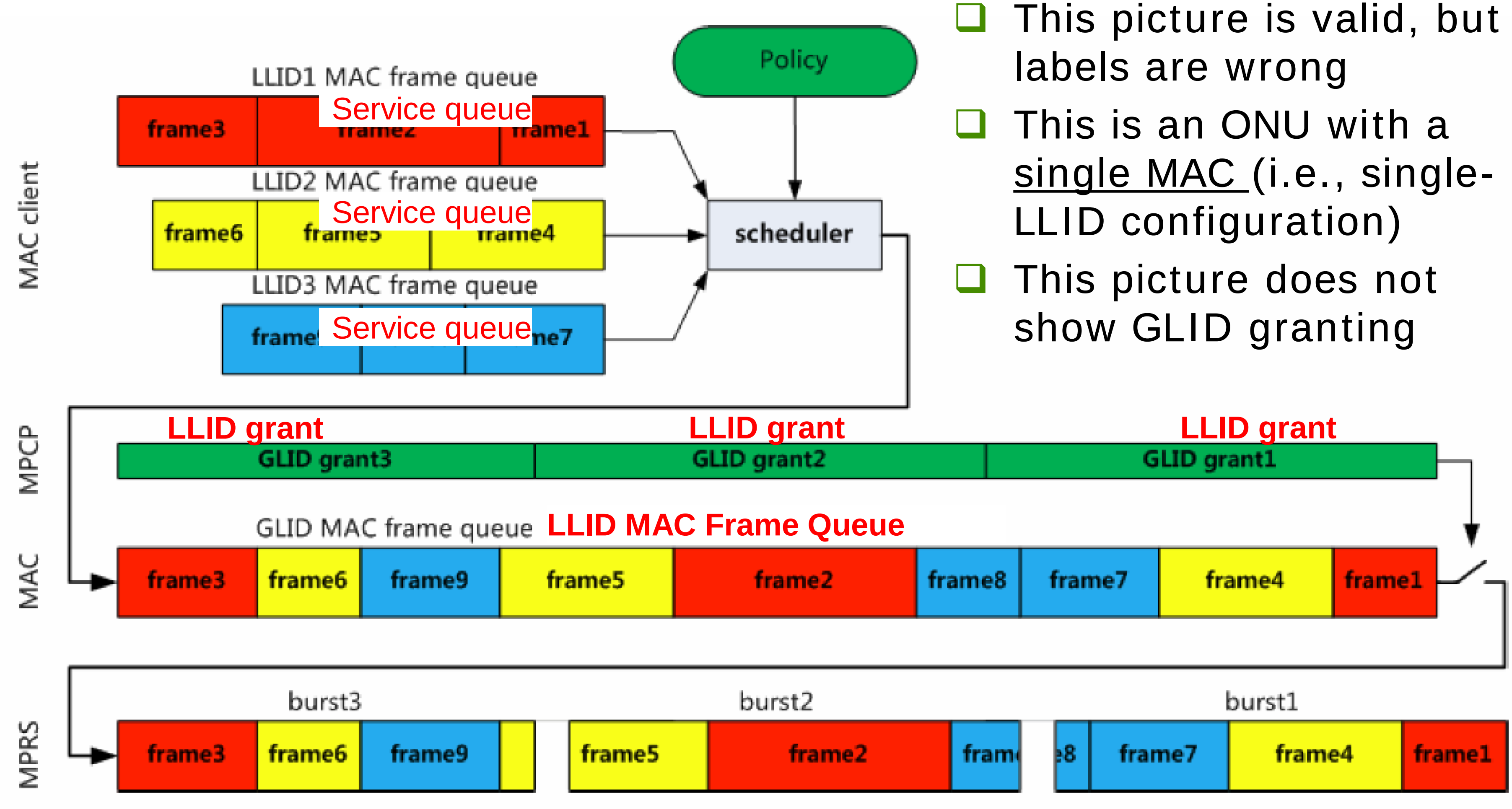
Reassembly Buffer and Working Mechanism – a Deeper Look

Glen Kramer, Broadcom

Need to use proper terms

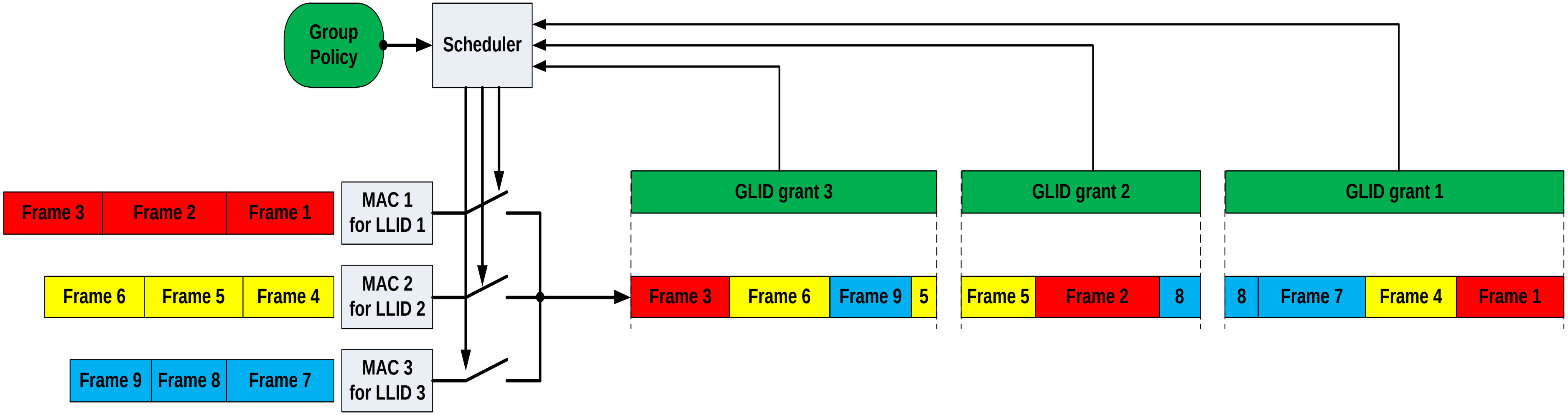
- According to definitions in .3ah, .3ah, and .3ca, LLIDs start and terminate at MAC entities at both ends of a link.



- This picture is valid, but labels are wrong
- This is an ONU with a single MAC (i.e., single-LLID configuration)
- This picture does not show GLID granting

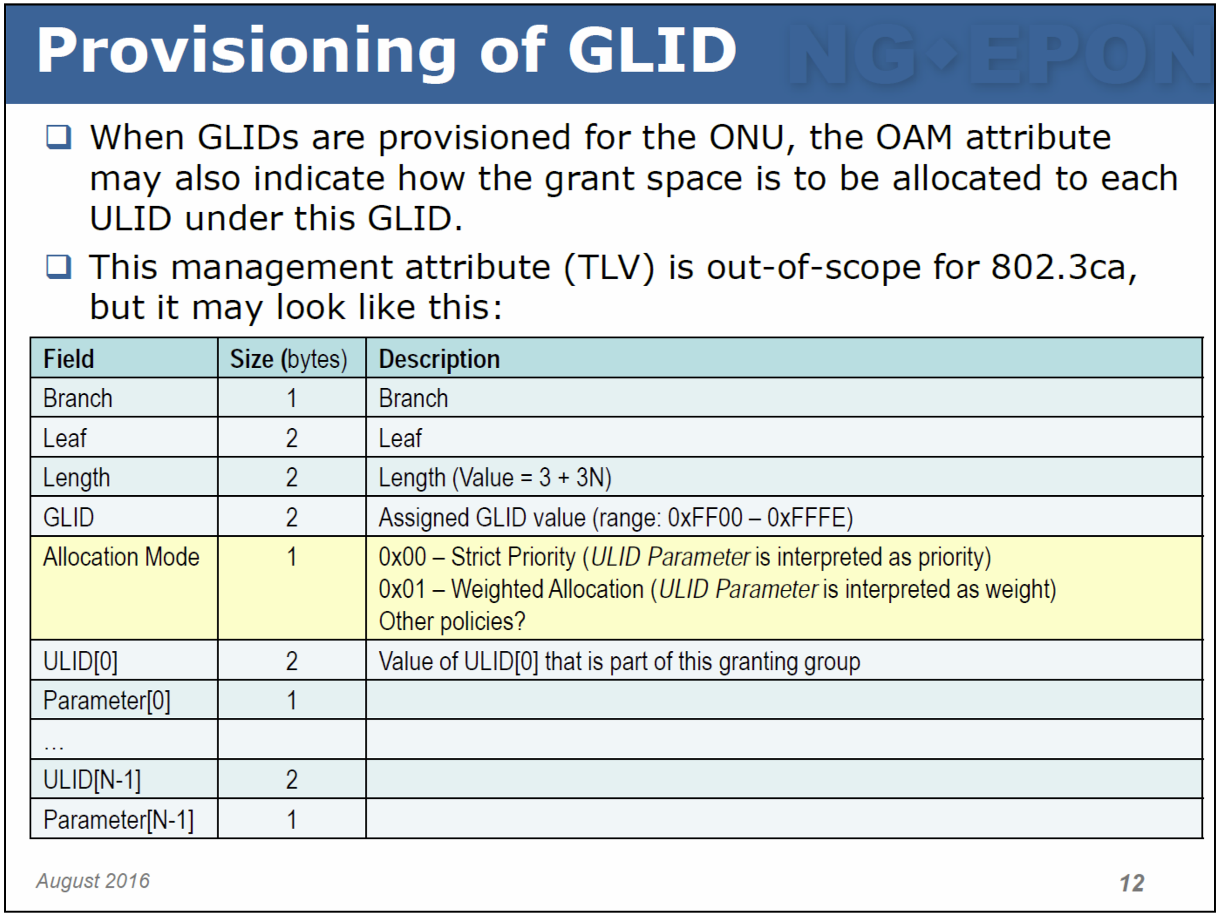
ONU with multiple LLIDs

- This is a more accurate picture of GLID granting in mL-ONU



Group Policy determines fragmentation

- ❑ In September we discussed a way to provision GLIDs via eOAM.
- ❑ A parameter called **Allocation Mode** determines how the grant length is distributed among the group member ULIDs.



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	

August 2016 12

Discussed during September meeting in Ft. Worth.

Note: eOAM is out of scope for 802.3ca.

Examples of Other Group Policies

Code	Name	Behavior
00	Strict Priority; Fragment None	• If any of the ULIDs in the GLID have fragments pending, they may be transmitted to complete the frame. • No new frames are allowed to be fragmented. • In some cases, a grant may have an unused remainder.
01	Strict Priority; Fragment Last	• Generally, under SP allocations, frames in the middle of the grant are never fragmented (as long as high-priority frame is available, no lower-priority ULID will transmit). • The last frame in a grant may be fragmented to fill the grant.
02	Weighed Allocation; Fragment None	• Pending fragments may be transmitted (Previously-fragmented frames may or may not get completed.) • No new frames are allowed to be fragmented. • Deficit counters may be used to determine which LLIDs get to transmit in each grant, and how much. • In some cases, a grant may have an unused remainder.
03	Weighed Allocation; Fragment Last	• Pending fragments may be transmitted. • No new frames are allowed to be fragmented, except the last frame in a grant.
04	Weighed Allocation; Fragment Any	• Any ULID in the group transmits the number of EQs solely determined by grant length and ULID's weight. • Any ULID may complete a frame and fragment a new frame.
	Other?	

Group Policies + Fragmentation Flag

Code	Policy Name	Frag. Flag	Behavior
00	Strict Priority	0	• If any of the ULIDs in the GLID have fragments pending, they may be transmitted to complete the frame. • No new frames are allowed to be fragmented. • In some cases, a grant may have an unused remainder.
		1	• Generally, under SP allocations, frames in the middle of the grant are never fragmented (as long as high-priority frame is available, no lower-priority ULID will transmit). • The last frame in a grant may be fragmented to fill the grant.
01	Weighed Allocation; Unit = Frame	0	• Pending fragments may be transmitted to complete frames. • No new frames are allowed to be fragmented. • Deficit counters may be used to determine which LLIDs get to transmit in each grant, and how much. • In some cases, a grant may have an unused remainder.
		1	• Pending fragments may be transmitted to complete frames. • If an entire frame fits in the grant, it should not be fragmented. • The last frame in a grant may be (or remain) fragmented.
02	Weighed Allocation; Unit = EQ	0	• Pending fragments may be transmitted, though not necessarily to complete a frame. • No new frames are allowed to be fragmented.
		1	• Any ULID in the group transmits the number of EQs solely determined by grant length and ULID’s weight. • Any ULID may complete a frame and fragment a new frame.

Conclusion

- The key idea in zhangweiliang_3ca_1a_0317.pdf is to have only one pending frame fragment per GLID instead of one pending fragment per each member ULID.

- <u>MPRS in D0.2 already supports the required behavior and no changes are required.</u>

- To achieve the operation described in zhangweiliang_3ca_1a_0317.pdf, all we need to do is to define a proper Group Policy in eOAM attribute (out-of-scope for 802.3ca).
  - Examples:
    - Strict Priority; Fragment Last
    - Weighed Allocation; Fragment Last

Thank You