On Configuring Logical Links in EPON

Glen Kramer, Teknovus, Inc.

1 Introduction

As Ethernet PON (EPON) systems move into mass deployment, Service Providers have begun in earnest to explore the utility of this technology for delivering services other than best-effort Ethernet transport. Adding these new services, which commonly include telephony, video delivery, and leased-line transport, have the potential of increasing substantially the revenue generated by an EPON deployment, and in fact this additional revenue often plays a deciding role in the business case for an EPON deployment. As a rule, these new services place additional, significant demands on the performance of an EPON system. The ability to strictly enforce Service Level Agreements (SLAs), which can include bandwidth guarantees and stringent delay and jitter tolerances will be an absolute requirement for multi-service EPON systems.

EPON uses the Multi-Point Control Protocol (MPCP), which employs REPORT (an upstream message from the ONU) and GATE (a downstream message from the OLT) control messages, to request and assign transmission opportunities on the PON. This is the basic mechanism that controls the flow of data on the PON, and it used by higher level functions for bandwidth allocation, ONU synchronization, and ranging. The entity that receives a GATE and responds with a REPORT is called a logical link and is identified by Logical Link IDentifier (LLID). The number of LLIDs per ONU is a design choice. Each LLID may have one or more queues for user data. The P802.3ah draft standard has adopted an asymmetrical format for these two messages, in the sense that REPORTs carry information about the states of individual queues within an LLID, while GATEs convey aggregate, upstream grants pertinent to all of the queues in an LLID. Clearly, the number of LLIDs instantiated on an ONU can have a profound impact on performance, and in fact is one of the most important design choices made in designing an EPON system.

In this white paper we investigate how this asymmetrical message format affects EPON's ability to guarantee performance for various services and enforce SLAs on a per-queue basis. Specifically, we look at whether it is possible to guarantee throughput and delay on a per-queue basis when a single LLID contains several queues. We find that EPON systems using single-LLID ONUs are in effect delegating to the ONUs the task of scheduling the various queues within the upstream time slot, thus requiring ONUs

to be SLA-aware and capable of traffic shaping. This approach results in a highly complex, non-robust, and inefficient solution. A superior solution is described, in which a LLID is assigned for each queue and all EPON intelligence is concentrated at the OLT. We show that with this second architecture dramatic improvements to the system's efficiency, flexibility, interoperability, and cost-effectiveness are obtained.

2 EPON Overview and Architecture

An EPON is a point-to-multipoint (PtMP) optical network with no active elements in the signals' path from source to destination. The only interior elements used in EPON are passive optical components, such as optical fiber, splices, and splitters. EPON architecture saves cost by minimizing the number of optical transceivers, central office terminations, and fiber deployment. We refer the reader to [1] for an in-depth description of EPON architecture.

All transmissions in an EPON are performed between a head-end called the Optical Line Terminal (OLT) and tail-ends called Optical Network Units (ONUs) (Figure 1). The ONU serves either a single subscriber (fiber-to-the-home) or multiple subscribers (fiber-to-the-curb or fiber-to-the-multi-dwelling-unit). In the downstream direction, EPON is a broadcasting media; Ethernet packets transmitted by the OLT pass through a 1:N passive splitter or a cascade of splitters and reach each ONU. An ONU filters packets destined to its users and discards the rest (Figure 1).

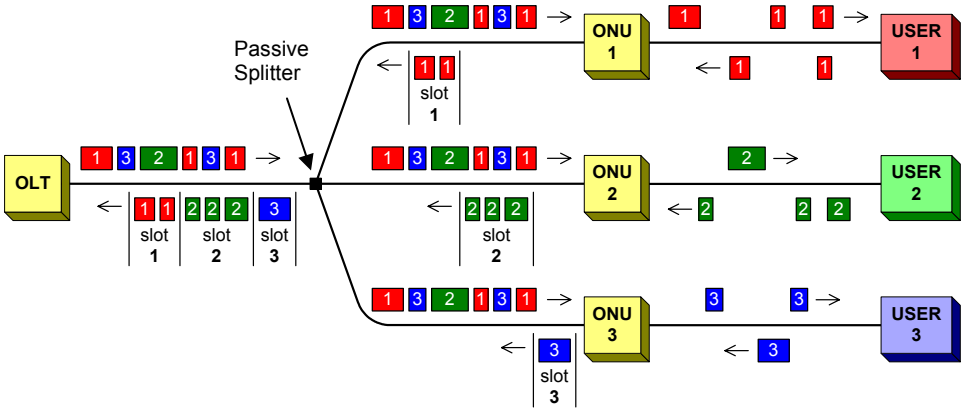


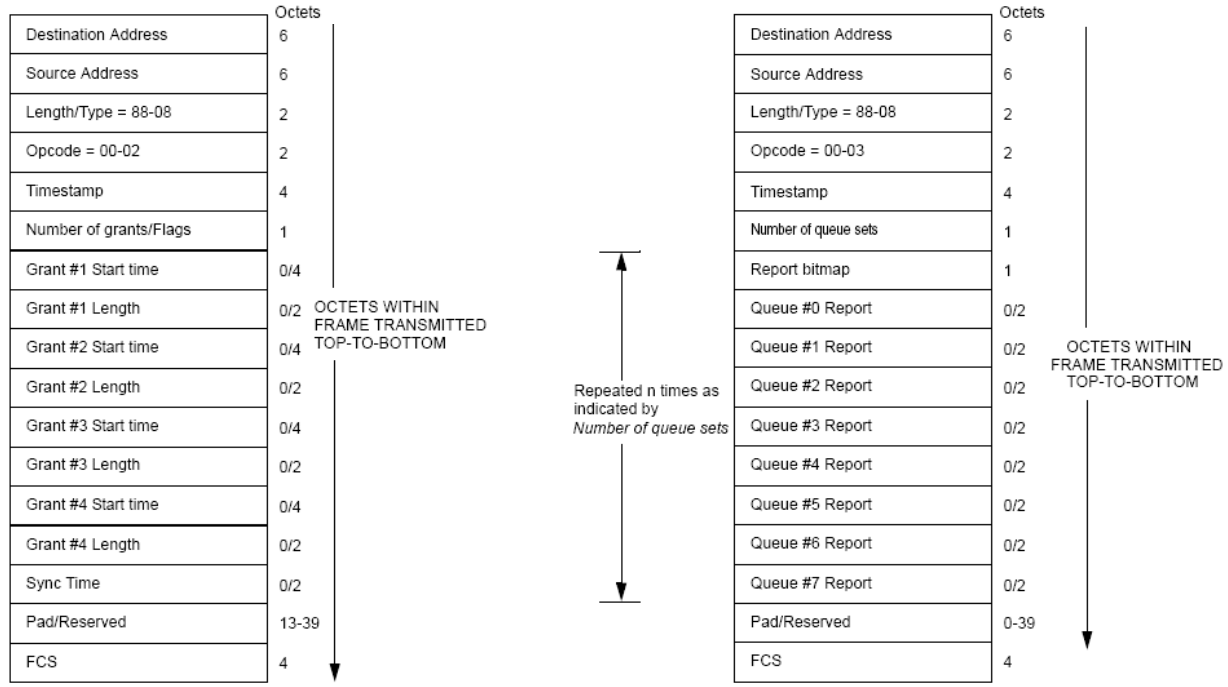
Figure 1: Upstream and downstream transmissions in EPON.

In the upstream direction (from the ONUs to the OLT), the ONUs need to employ some arbitration mechanism to avoid data collisions and fairly share the channel capacity. This is achieved by the OLT allocating (either statically or dynamically) non-overlapping, variable-sized transmission windows (timeslots) to each ONU. To enable timeslot assignment, the IEEE 802.3ah task force is developing a Multi-point Control Protocol (MPCP). MPCP uses two MAC control messages: GATE and

REPORT*. The GATE message is sent from the OLT to an ONU and is used to assign a timeslot to the ONU. The REPORT message is sent from an ONU to the OLT to request another timeslot by reporting the amount of queued data. Each message is a standard 64-byte MAC Control frame. MPCP is only a message-exchange protocol; IEEE 802.3ah specifically does not specify any algorithm for bandwidth allocation.

2.1 MPCP Message Format

The format of the GATE message is shown in Figure 2.a. A GATE message conveys up to 4 grants (transmission slots) to an ONU. In case the ONU has more than one backlogged queue, it is up to the ONU to decide how to allocate the grant between various queues.



(a) format of GATE message **(a) format of REPORT message**

Figure 2: Format of GATE and REPORT messages

The format of the REPORT message is shown in Figure 2.b. A REPORT message conveys status of up to 8 queues to the OLT. Each queue may report multiple thresholds, such that OLT may allocate grant size based on one of the thresholds, thus avoiding bandwidth loss due to slot under-utilization

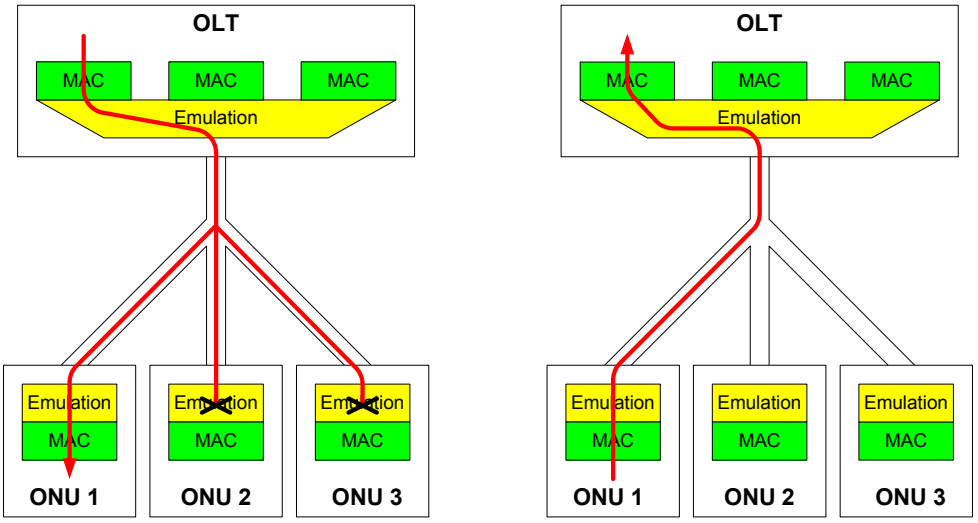
* Additional messages defined by the MPCP are used by the initialization process.

(packet delineation overhead). The number of thresholds that REPORT message can deliver depends on how many queues are to be reported. Given the 64-byte limit on the REPORT message, an ONUthat reports 8 queues may have up to up to 2 thresholds per queue. If only one queue is available at the ONU, it may report up to 13 thresholds.

2.2 Logical Links

To ensure compatibility with the IEEE 802 architecture, EPON employs a point-to-point emulation mechanism, which makes the EPON medium behave as a collection of point-to-point links. Emulation mechanisms rely on tagging Ethernet frames with a unique value called the Logical Link ID (LLID).

To allow point-to-point emulation, the OLT must have *N* MAC ports (interfaces), one for each logical link (Figure 3). When sending a frame downstream (from the OLT to an ONU), the emulation function in the OLT will insert the LLID associated with a particular MAC port on which the frame arrived (Figure 3). Even though the frame will be delivered to each ONU, only one ONU will match that frame's LLID with its own assigned value, and thus accept the frame and pass it to its MAC layer for further verification. MAC layers in all other ONUs will never see that frame. In this sense, it appears as if the frame was sent on a point-to-point link to only one ONU.



(a) downstream transmission (b) upstream transmission

Figure 3: Point-to-point emulation in EPON.

In the upstream direction, the ONU will insert its assigned LLID in the preamble of each transmitted frame. The emulation function in the OLT will de-multiplex the frame to the proper MAC port based on the unique LLID (Figure 3.b).

3 Objectives of EPON Scheduling Algorithm

In a subscriber access network, an ONU may serve one or more subscribers and can have one or more queues assigned to each subscriber. Different queues belonging to one subscriber can be used, for example, to serve different classes of traffic (i.e., voice, video, and data) with different quality-of-service (QoS) guarantees. To satisfy the network requirements, an EPON scheduler should meet the following objectives:

Guarantees: Unlike enterprise LANs, access networks serve non-cooperative users; users pay for service and expect to receive their service regardless of the network state or the activities of the other users. Therefore, the network operator must be able to guarantee a minimum bandwidth and maximum delay (latency) to each queue. Different services (queues) require different parameters. For example, voice service requires a delay bound of 1.5 ms [2], but needs only fixed and small bandwidth. Video traffic requires variable bandwidth, but can tolerate larger delay.

Fairness: Idle queues should not consume any bandwidth. Excess bandwidth left by idle queues should be redistributed among backlogged queues in a fair and predictable manner, for example, in proportion to weights assigned to each queue. The fairness of bandwidth distribution should be preserved regardless of whether the queues are located in the same ONU or in different ONUs.

Protection: Misbehaving user or application should not be able to disrupt services of other users or applications. For example, if one subscriber generates a large volume of high priority traffic, an EPON scheduler should be able to effectively isolate and limit this particular subscriber.

Robustness: A discrepancy may occur between OLTs knowledge of ONUs' states and actual ONUs' states, say, due to lost GATE or REPORT message. In case such a discrepancy occurs, the scheduling algorithm should not fatally fail. It should continue to function and return to its normal and efficient operation upon discrepancy removal.

4 One LLID per ONU vs. Multiple LLIDs per ONU

EPON networks can be configured in a number of ways. For example, a single logical link can be allocated to each ONU, or a single logical link can be allocated to individual queues in each ONU. Below

we consider the advantages and disadvantages of each of these two approaches.

4.1 One logical link per ONU

In this configuration, one logical link is allocated to the entire ONU. In effect, this presents a hierarchical scheduling structure, where a top-level scheduler in the OLT assigns an aggregated slot to an ONU; this slot is further subdivided between multiple queues by a low-level scheduler in the ONU (Figure 4).

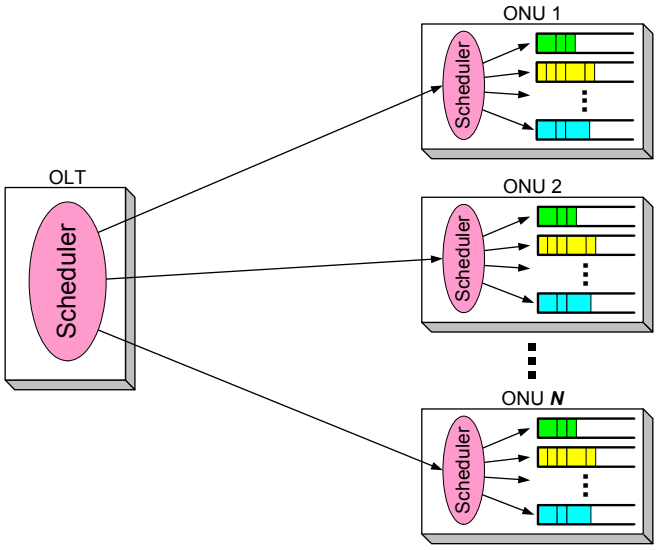


Figure 4: One logical link per ONU (hierarchical scheduling).

The main advantage of this scheme is reduced bandwidth consumption by MPCP messages needed to schedule ONUs transmissions. The question, however, is whether the hierarchical scheduling scheme could meet the objectives stated in section 3.

We investigate performance of hierarchical scheduling scheme by first considering a low-level scheduler implementing a simple *strict (exhaustive) priority queuing*. In exhaustive priority queuing schemes, a queue is served only when all higher-priority queues are empty.

4.1.1 Preemptive priority queuing in ONU

It is generally known that in priority queuing, low-priority queues could starve. In [3] it was shown that in EPON, the lower-priority queues starve even under light load. This is caused by the fact that during the time lag between ONU's reporting queue states and arrival of the corresponding grant (i.e., between sending a REPORT and transmission of the reported data), more packets arrive to the queues.

Newly-arrived packets may have higher priority than some packets already stored and reported to the OLT. These new packets preempt lower-priority packets and will be transmitted in the next transmission slot. Since the new packets were not reported to the OLT, the given slot cannot accommodate all the stored packets. This leads to some lower-priority packets being left in the queue. This scenario may repeat many times, resulting in some lower-priority packets being delayed for multiple cycle times. The lower the queue priority is, the higher the starvation probability for this queue.

Another problem is caused by changed packet delineation bounds. The delineation alignment changes, because the transmitted packets are different from the reported packets. Since Ethernet packets cannot be fragmented (while also complying with IEEE 802.3), packet preemption results in an *unused slot remainder* (unless added higher-priority packets have the same total size as preempted lower-priority packets, which is rare).

4.1.2 Non-preemptive priority queuing in ONU

The problem of queue starvation at light loads can be mitigated by using a *non-preemptive* priority queuing in ONUs. In non-preemptive queuing, ONUs can transmit only previously-reported packets, even if more higher-priority packets arrive after the last REPORT was sent. Non-preemptive queuing can be implemented, for example, as a two-stage buffer [3] or using pointers to the last reported packet in each queue.

Non-preemptive queuing solves the problem of packet delineation overhead and ensures that the unused remainder is zero if the OLT grants slot size based on the reported threshold[†]. It also solves the problem of lower-priority queue starvation under the light load.

However, a shortcoming of non-preemptive queuing is an increased queuing delay, since all packets will have to wait a full cycle between being reported and being transmitted. Table 1 shows average and maximum packet delays for higher-priority packets with a 1 ms cycle (interval between ONUs transmissions).

	Preemptive queuing	**Non-preemptive queuing**
Average queuing delay (ms)	0.5 ms	1.5 ms
Maximum queuing delay (ms)	1.0 ms	2.0 ms

Table 1: Queuing delay of high-priority packets with 1 ms cycle.

[†] A complication arises when multiple thresholds per queue are reported. In this situation, it is very difficult for ONU to find out which particular combination of thresholds has resulted in the given total slot size.

The increased delay in a non-preemptive priority queuing scheme could become a problem. All high-priority packets such as system alarms, failure indication, etc. may have to endure a longer delay. For example, consider the delay budget for voice traffic. ITU-T Recommendation G.114 "*One-way transmission time*" specifies 1.5 ms one-way propagation delay in access network (digital local exchange). To keep the maximum delay within this bound, the cycle time will have to be reduced to 750 ns. This, in turn increases guard band and scheduling overheads [4].

Even more serious problem in a non-preemptive priority queuing scheme is that *OLT has no means to limit misbehaving high-priority queue without completely starving all lower-priorities queues*. To fix this problem and provide service protection from misbehaving applications, ONU should implement some kind of ingress shaping on a per-queue basis. However, considering that available bandwidth depends on the state of all the ONUs in EPON, such ONU ingress shaping function should have a global view of the EPON. Without such global feedback, ingress shapers have no choice but to trim incoming traffic to minimum rates guaranteed to each queue, even if excess bandwidth is available in the EPON. Not being able to utilize the excess bandwidth eliminates *multiplexing gain* – one of the main advantages of EPON over alternative subscriber access architectures. Furthermore, absence of a standard protocol to control parameters of ingress shapers will necessitate proprietary solutions with detrimental effect on interoperability of EPON devices.

4.1.3 Rate-proportional schedulers

An alternative solution may be to use any of several available fair-queuing or proportional-rate schedulers in EPON: weighted fair queuing (WFQ) [5], worst-case fair weighted fair queuing (WF^2Q) [6], virtual-clock fair queuing (VCFQ) [7], self-clocked fair-queuing (SCFQ) [8], start-time weighted fair queuing (STFQ) [9], weighted round robin (WRR), deficit round-robin, carry-over round robin, and many others. Such schedulers would allocate each queue a fixed fraction of the total bandwidth (or fraction of a slot) available to an ONU. Generic algorithms can even be enhanced to allow separate control over guaranteed bandwidth and excess bandwidth for each queue.

Nevertheless, the schemes employing rate-proportional schedulers in the ONU are not free from problems. One notable problem is that there is no way for OLT to foresee how many packets each queue would transmit, unless the OLT knows exactly all packet sizes in each queue. Often, the number of packets transmitted by a queue would not correspond to the threshold reported in the REPORT message, especially if the queue is allowed to use excess bandwidth. This discrepancy results in the situation that an assigned slot will have an unused remainder, even if the OLT chooses a slot size for an ONU exactly equal to a sum of thresholds reported by the ONU.. A formula to determine the estimated size of the

unused remainder for an arbitrary packet-size distribution was derived in [10]. With the empirical tri-modal packet-size distribution reported in [11], the average size of the remainder is 595 bytes. Total bandwidth lost due to unused remainders in an EPON with 32 ONUs and 1 ms cycle time approximately equals 152 Mbps.

Another significant problem is that fairness objective cannot be met. Fairness in bandwidth allocation will only be achieved between queues of the same ONU, but not between queues located in different ONUs. Two queues with identical SLAs and identical backlog of packets may get different service.

And finally, the queue arbitration decisions are delegated to the ONUs. Yet, there exists no universal protocol to setup per-queue parameters in the ONU. As vendors may use different intra-ONU scheduling algorithms, so the sets of parameters required would be different. In addition to resulting in more complicated and costly ONUs, this problem would decrease interoperability and complicate testing.

Scheduling schemes where only one logical link is allocated per multiple queues (per ONU) cannot meet EPON objectives of providing SLA guarantees, service fairness and protection.

4.2 One logical link per queue

The simplest and most robust solution is to allocate a single logical link to each queue (Figure 5). This will eliminate any need for low-level scheduler or ingress shapers in the ONU and will concentrate all the intelligence in the OLT.

The OLT will receive a separate REPORT message from each individual LLID representing just one queue. Since the OLT issues a separate GATE message for each LLID, it can easily limit one queue, while giving more excess bandwidth to another queue. The ONU in this case becomes very simple.

In should be noted that this approach could have higher scheduling overhead, since a separate message now would be required for each queue, instead of one message per ONU. Considering an EPON system with 32 ONUs, control message overhead would increase by approximately 150 Mbps. Yet, considering that this scheme eliminates packet delineation overhead equal approximately 152 Mbps as was shown above, overall performance remains approximately the same. If ONUs are equipped with less than 8 queues, an LLID-per-queue scheme has even smaller overall overhead than LLID-per-ONU scheme.

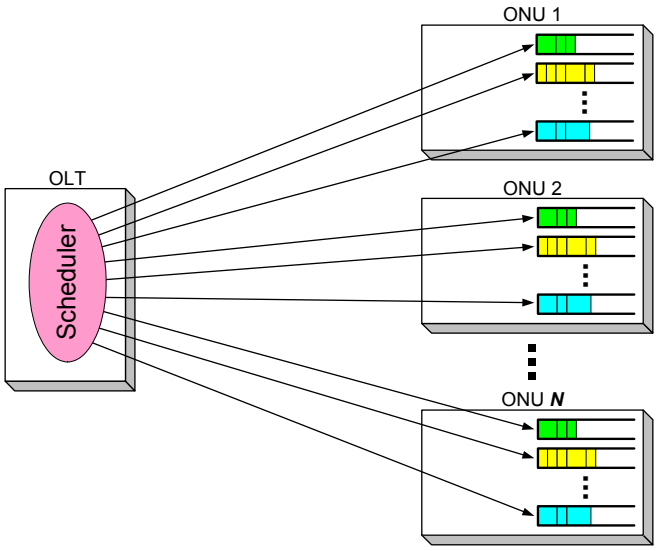


Figure 5: One logical link per queue (single-level scheduling).

Having independent GATEs per queue allows ONU to have different cycle times (polling intervals) for different queues. For example, data traffic such as FTP transfer is not delay-sensitive, and overall performance would increase if data queues are served less often, but are given large slots. Voice traffic, on the other hand, has low volume and is very delay sensitive. It makes sense, therefore, to poll voice queues frequently but with smaller slots.

We also note that allocating an independent logical link per queue does not increase total guard band overhead. As is shown on Figure 6, the OLT can schedule grants to same-ONU LLIDs very close to each other, such that the Data Detector function located in the common PHY will not shut down the laser between two such grants. In effect, transmissions from multiple LLIDs in one ONU will be concatenated together and will look like just one grant. Thus, the number of guard bands would remain the same as in the case of a single logical link allocated to the ONU.

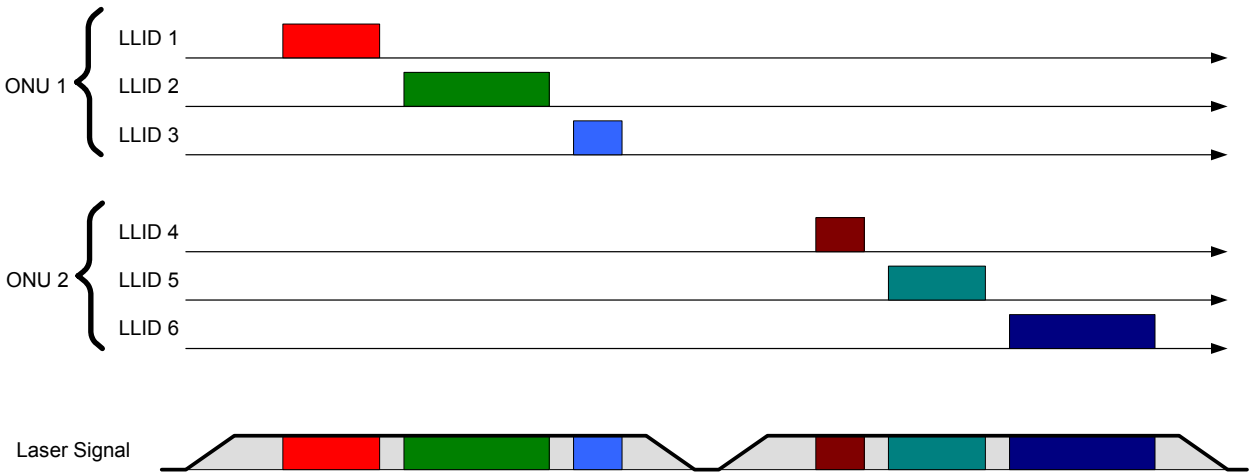


Figure 6: Grants from multiple LLIDs in one ONU are concatenated.

5 Conclusions

EPONs use multi-point control protocol (MPCP) to assign transmission opportunities to multiple ONUs. MPCP relies on GATE and REPORT messages. While the REPORT message carries information about individual queues in the ONU, the GATE only assigns one aggregated slot to the ONU. In effect, this delegates the task of scheduling various queues within the slot to the ONUs, requiring ONUs to be SLA-aware and to be able to perform traffic shaping functions. This approach results in a highly complex, non-robust, and inefficient solution. An alternative approach that assigns a separate logical link for each queue allows full EPON intelligence to be concentrated at the OLT, thus improving EPON's efficiency, flexibility, interoperability, and cost-effectiveness.

In this paper we focused on a single-user-per-ONU or fiber-to-the-home (FTTH) scenario. If EPON is deployed in a fiber-to-the-curb (FTTC) or fiber-to-the-multi-dwelling-unit (FTTMDU), i.e., if one ONU serves multiple independent users, the requirement to have multiple LLIDs allocated to the ONU becomes even more important. For one reason, the OLT (operator) should have a centralized control over each user access and SLA. Another reason is that usage statistics, such as lost or corrupted frames, throughput, etc. must be maintained per subscriber.

References

[1] G. Kramer, B. Mukherjee, and A. Maislos, Ethernet Passive Optical Networks, In S. Dixit, editor, IP over WDM: Building the Next Generation Optical Internet, John Wiley & Sons, Inc., February 2003.

[2] ITU-T Recommendation G.114, One-Way Transmission Time, in Series G: Transmission Systems and Media, Digital Systems and Networks, Telecommunication Standartization Sector of ITU, May 2000.

[3] G. Kramer, B. Mukherjee, S. Dixit, Y. Ye, and R. Hirth, "*Supporting Differentiated Classes of Service in Ethernet Passive Optical Networks*", Journal of Optical Networking, vol. 1, no. 8/9, pp. 280-298, August 2002.

[4] G. Kramer, "*How efficient is EPON?*", white paper, available *at* www.ieeecommunities.org/epon

[5] A. Demers, S. Keshav, and S. Shenker. "*Analysis and simulation of a fair queueing algorithm,*" Journal of Internetworking Research and Experience, pp. 3-26, October 1990.

[6] J. C. R. Bennett and H. Zhang, "*WF2Q: Worst-case Fair Weighted Fair Queueing,*" Proceedings of INFOCOM '96, pp. 120-128, San Francisco, March, 1996.

[7] L. Zhang, "*Virtual clock: a new traffic control algorithm for packet switching networks,*" Proceedings of ACM SIGCOMM'90, pp. 19-29, September 1990.

[8] S. J. Golestani, "*A self-clocked fair queuing scheme for broadband application,*" Proceedings of IEEE INFOCOM'94, pp. 636-646, Toronto, Canada, June 1994.

[9] P. Goyal, H. M. Vin, and H. Cheng, "*Start-time Fair Queuing: A Scheduling Algorithm for Integrated Services Packet Switching Networks,*" IEEE/ACM Transactions on Networking, vol. 5, no. 5, pp. 690-704 October 1997.

[10] G. Kramer, B. Mukherjee, and G.Pesavento, "*Ethernet PON (ePON): Design and Analysis of an Optical Access Network,*" Photonic Network Communications, vol. 3, no. 3, pp. 307-319, July 2001.

[11] D. Sala and A. Gummalla, "*PON Functional Requirements: Services and Performance,*" presented at IEEE 802.3ah meeting in Portland, OR, July 2001. Available at http://grouper.ieee.org/groups/802/3/efm/public/jul01/presentations/sala_1_0701.pdf.