Low Latency DOCSIS: Concepts And Experiments

A Technical Paper prepared for SCTE•ISBE by

Tushar Mathur
Staff Systems Engineer, CTO Office
CommScope Inc.
90 Matheson Blvd. W., Mississauga, ON, Canada
tushar.mathur@commscope.com

Ram Ranganathan
Director of Systems Engineering, CTO Office
CommScope Inc.
90 Matheson Blvd. W., Mississauga, ON, Canada
ram.ranganathan@commscope.com

Greg Gohman, CommScope Inc.

Bob Zhang, University of Waterloo

Table of Contents

Title **Page Number**

List of Figures

Title **Page Number**

List of Tables

Title **Page Number**

1. Introduction

Today's internet traffic typically comprises data, voice, or video traffic with no extraordinary means to logically segregate traffic based on its network latency sensitivity. Applications have varying requirements for bandwidth, latency or jitter. Some apps require high bandwidth, such as large file downloads or video traffic, and certain apps require low latency, such as online gaming traffic or high frequency trading. The online gaming industry is on a rapid growth path and has become an exciting mainstream revenue source. With an increasing demographic that streams gameplays, the cloud gaming services are bringing new online gaming experience closer to the consumers and it will require support from the 10G initiative driven MSOs to deliver the best quality of experience by ensuring *high* bandwidth and *low* latency or *low* lag support. By enhancing the user experience, the MSOs have an opportunity to generate a new revenue stream. Welcome to the world of Low Latency DOCSIS!

The LLD architecture as proposed by CableLabs enables a logical separation of the latency sensitive non-queue building traffic and regular queue-building internet traffic in to two separate queues. The two queues, Low Latency SF and Classic SF are encapsulated in an Aggregate Service Flow (ASF) to shape the traffic. A key innovation that is part of the LLD architecture is a new scheduling service known as Proactive Grant Scheduling (PGS) [1].

There are multiple sources of latency in DOCSIS networks, including protocol/application dependent queuing delays, propagation delay, Request-Grant delay, channel configuration (OFDM or SC-QAM interleavers, cyclic prefix, FEC, etc.), and switching/forwarding delays. The purpose of LLD is to reduce latency from two of these sources – protocol/application dependent queuing delays and Request-Grant delays.

This paper will focus on the LLD architecture basics and experimental results from the lab studies using the concept of an LLD ASF and PGS in the *DOCSIS Upstream*. The paper will also compare LLD capable system latency with classic latency.

2. Low Latency DOCSIS Architecture And Goals

Often times, the bandwidth or speed of a connection is confused with latency of an application. Bandwidth or speed of a connection means how much of the data can be downloaded or uploaded within a time interval. For example, to watch a 4K YouTube video or to download a Call Of Duty 25 GB game patch, a user would need a good bandwidth. There are times when the bandwidth of a connection isn't enough to deliver the best QoE. For example, a multiplayer game of Call Of Duty requires players to shoot at other players as well as download any real time rendering of dynamic game environment. This typically results in packets being transmitted at a bit rate of 100 kbps to 200 kbps in the Upstream and Downstream direction. It is important that the packets reach their destination as quickly as possible so that the player does not get shot themselves first and the game environment rendering is synced with a player's action. If the gaming environment actions are not synced with a players action then it is because of a "high lag" in the network. This time duration of the packets to reach the Call Of Duty gaming server and returning a response to the multiplayer gamer is called Latency. So, to deliver the best QoE it is important to maintain reasonable bandwidth and latency. Inside a home there are multiple users transmitting traffic in the US and DS. Some of the internet traffic may be file download, or a YouTube video, or a Netflix video and other may be gaming traffic or video conferencing.

Typically all the traffic will flow in to a single DOCSIS service flow, with a mix of traffic that builds queues like the video streaming apps and other traffic that doesn't build queues like a multiplayer gaming app. The problem with this architecture is that the gaming app gets treated similar to a video streaming app, appending non-queue building traffic in to queue building traffic. Hence, this adds latency & jitter to an already latency sensitive application.

The LLD architecture uses a logical construct called an Aggregate Service Flow (ASF) that encapsulates two underlying service flows – one for Non-Queue building traffic and the other one for Queue building traffic. The intention of separating application's traffic in to two logical queues is to make sure that the application data that builds queues in the DOCSIS access network don't cause delays for data that does not build queues.

For example, let us assume there are two applications that are transmitting traffic in the DOCSIS channel.

1. Online gaming application traffic is typically a few hundred kbps of UDP payload and will not cause queues to build in a SF. UDP does not have any congestion algorithms and will transmit at a set rate, without retransmissions. We call this traffic as the NQB or Low Latency traffic

2. Large file upload traffic in the order of a few MB of TCP payload and will cause formation of queues in a SF. The inherent nature of TCP congestion algorithms is to seek as much bandwidth as possible, cause retransmissions in case of packet loss which results in formation of queues. We call this as QB or Classic Traffic

As compared to a large file upload, the gaming traffic requires the best latency possible so that a gamer can have the best Quality of Experience. If both the traffic types i.e. gaming & file upload are contending in the same SF, although the UDP traffic is so low in bandwidth, it gets congested with the TCP traffic that causes queues to build-up. In other words, the a latency sensitive application gets delayed in the SF queue by a queue building application. So, by using LLD, the latency sensitive NQB application gets its own SF queue without impacting the latency of other QB applications.

Now, let's take a look closer look at the features provided by the LLD architecture. Figure 1 shows the LLD architecture and its components. The CMTS has many important functions in defining the QoS in LLD:

1. A new ASF encapsulating LL and CL SF, known as the LLD ASF
2. A new Weighted Round Robin Scheduler for the two SFs
3. Rate shaping these two SFs at an aggregate level
4. A new scheduling type known as the Proactive Grant Service
5. Traffic Classification in to SFs using DSCP and ECN fields in the IP header
6. Active Queue Management – a new AQM algorithm called the Immediate AQM and Coupled AQM for the two constituent Service Flow
7. Queue Protection
8. Latency Histograms

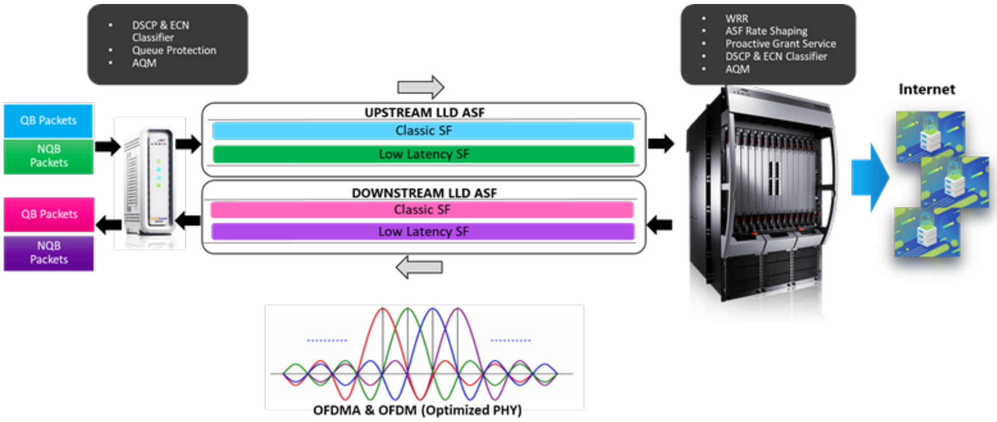


Figure 1 LLD Architecture

The LLD ASF provides an encapsulation to the traffic shaping of the LL SF and CL SF by enforcing an Aggregate Maximum Sustained Rate (AMSR). LL SF and CL SF are not like traditional service flows that are shaped by an MSR value. There are always only two SFs in the LLD ASF. For example, if the ASF AMSR is set to 100 Mbps then the traffic flowing in to LL and CL SF can be 20 Mbps and 80 Mbps or it can be 30 Mbps and 70 Mbps respectively, or in any other proportion but bounded by the ASF AMSR. In the upstream LLD ASF, the traffic flowing in the individual SFs will depend on the number of grants each service flow receives.

The Granting mechanism in LLD is governed by a Weighted Round Robin Inter-SF Scheduler running on the CMTS. The scheduler is to behave with conditional priority of providing LL SF grant priority without starving the CL SF. The WRR weight is a configurable parameter with a maximum value of 255. A 230 value set in the LL SF would mean 90% weight (230/255) of grants is to be provided to the LL SF. The weights provided by scheduler does not result in an unfairness between the two service flows because of the Coupled AQM feature of the LLD architecture. More on that later in this section.

Traditionally, the DOCSIS upstream data transmission follows the mechanism described in Figure 2. As soon as a packet arrives, a Bandwidth (BW) Request (REQ) is transmitted to the CMTS to allocate bandwidth requested by the CM. CMTS responds to the CM by sending a Bandwidth Grant (GNT) to the modem based on QoS parameters governed by the CMTS. Once the modem receives the Bandwidth Grant in a MAP packet, it will process the MAP and transmit the data packet to the CMTS, from where it is routed towards a server in the internet. The time for the entire process is given by Δ1 + Δ2 + Δ3, where Δ1 = Packet Processing Delay at the CM + Waiting for a REQ transmission slot + US propagation delay of BW REQ; Δ2 = BW REQ processing delay at the CMTS + MAP generation delay + DS propagation delay + Wait Time for GNT; Δ3 = US propagation delay of actual data to be transmitted

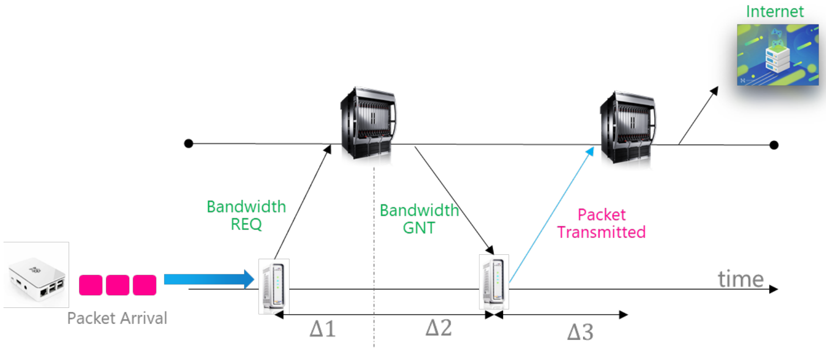


Figure 2 Traditional DOCSIS REQ-GNT Cycle

LLD introduces a new data scheduling type known as the Proactive Grant Service or PGS. PGS enables a faster request grant cycle by eliminating the need for a BW REQ even though it’s not prohibited to send a BW REQ on a PGS SF. CM will still be able to send BW REQ if the SF bandwidth demands are not met by PGS alone. In PGS, BW GNTs are continuously sent to the modem at a Guaranteed Grant Rate (GGR in bps) and at a Guaranteed Grant Interval (GGI in microseconds) value as soon as some activity is detected by the CMTS’s proprietary Activity Detection algorithm. GGI is the interval between successive data transmission opportunities. CMTS can track the bandwidth utilization and can adjust the GGR depending on any anticipated future demands, but at the time of writing this paper, the current CMTS implementation does not adjust GGR. If there is no activity detected on the SF then PGS will switch to sending unicast request opportunities at Guaranteed Request Interval (GRI in microseconds). The CMTS traffic shaper makes sure that the SF is not getting more grants than its maximum sustained rate by verifying the bounds of GGR and GGI. GGR, GGI, and GRI are configurable values in a PGS enabled flow. Figure 3 shows that with PGS, the upstream transmission time is shortened to Δ2’ + Δ3 where Δ2’ includes DS propagation delay and a reduced processing time for MAP generation. It is typically less than or equal to the GGI. If GGI is set to a value less than Δ1 + Δ2, PGS will provide a reduction in the upstream transmission time.

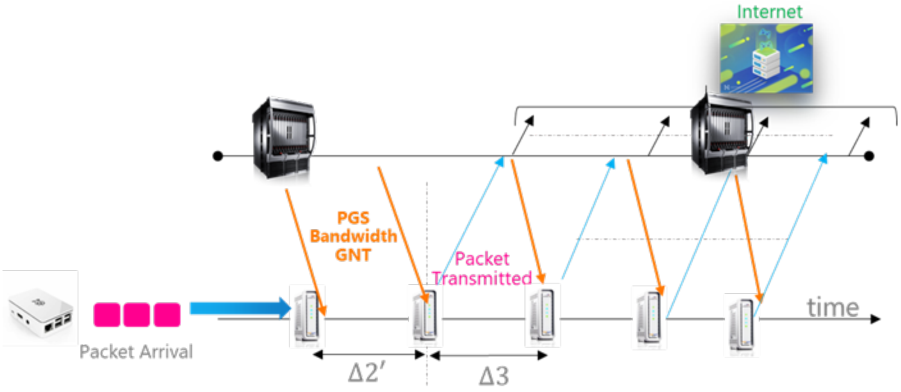
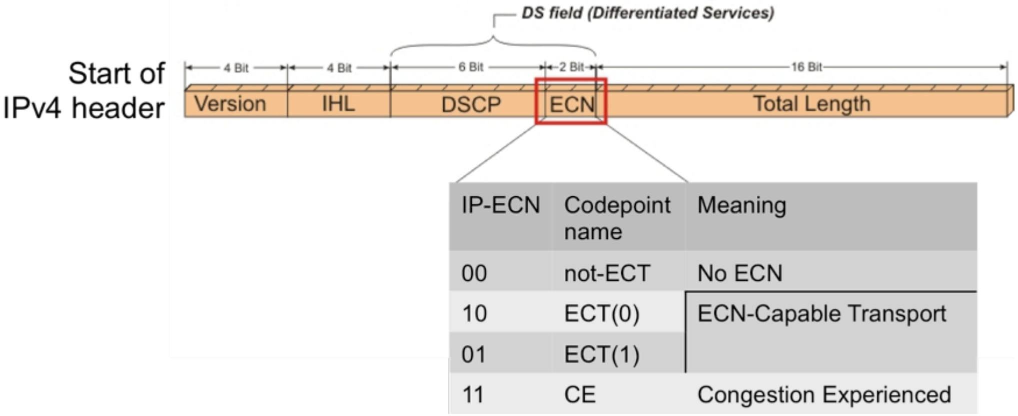


Figure 3 PGS Granting Mechanism

In LLD, packet classification plays an important role in placing a packet into a particular SF – CL or LL. The classifiers in LLD examine the 8 bit Differentiated Services field in the IP header for DSCP (MSB 6 bits) and ECN value (LSB 2 bits). For example, the classification can be made based on a packet’s DSCP field marked as EF or ECN field is set to ECT(1) or CE (see figure 4) then it will get mapped to the LL SF and any other traffic will be transmitted in the CL SF by default.



IP-ECN	Codepoint name	Meaning
00	not-ECT	No ECN
10	ECT(0)	ECN-Capable Transport
01	ECT(1)	
11	CE	Congestion Experienced

Figure 4 Differentiated Services Byte in IP Header

The Active Queue Management algorithms run on CL and LL SFs, both. The CL SF is to use the DOCSIS PIE algorithm, that drops packets as the queues build, to maintain a target latency defined by the configuration. The LL SF will implement a new AQM algorithm known as Immediate AQM, that does not drop packets but marks them with ECN bits. Default marking starts at 475 µs and will always mark the packets beyond 1 ms of latency. As stated earlier, the AQMs act as coupled AQM on both of the constituent SFs i.e. the IAQM of LL SF is coupled to the DOCSIS PIE of CL SF. The coupling will act as a backstop on the LL SF if the CL SF is getting overwhelmed by traffic. When the CL SF throughput is overwhelming the system with requests for the grants to keep up, the ECN marking is induced in the LL flow. The induced ECN marking in the LL SF reduces the bandwidth demands in the LL flow and the remainder grants from the ASF token bucket will be available for the CL flow.

Queue Protection categorizes packets into the application data flows, termed Microflows. All packets of each Microflow are characterized by identical values in a set of header fields. QP algorithm must act on every Microflow that becomes a queuing source in the LL SF. If the LL SF buffers are getting filled at a critical threshold then that queuing source needs to be redirected in to the CL flow.

3. Experimental Setup

There are some delays associated with the Upstream and Downstream PHY layer of DOCSIS. These delays can be minimized depending on an MSOs network conditions because there are trade-offs between channel robustness vs latency. For the experiments in this paper, the following PHY and MAC layer settings were configured.

In the upstream, an OFDMA channel with symbols per frame (k) set to 16 with 50 kHz subcarriers. The Cyclic Prefix value = 0.9375 µs and the Rolloff Period = 0.3125 µs were set for the channel to reduce PHY latency. The channel width of the OFDMA channel was set to 42 MHz. With 1K QAM, this resulted in a channel capacity of 336.60 Mbps.

In the downstream, a 192 MHz wide OFDM channel was configured. Cyclic Prefix = 0.9375 µs and Rolloff Period = 0.625 µs. The time interleaver depth = 1 (Depth of the time interleaver in symbols of an OFDM channel). With 4K QAM, the resulting downstream channel capacity is approximately 2100 Mbps.

There are some delays associated with the MAC layer of DOCSIS. These delays can be minimized depending on an MSOs network conditions. Values of these parameters can be adjusted to further reduce DOCSIS MAC latency.

MAP-size of the channel can be adjusted between 1 to 13. MAP-size is configured as an average size in 800 microseconds ticks. A setting of map-size = 1 implies 800 µs. Lowering the MAP-size results in faster acquisition of a bandwidth slot. For these experiments, the MAP-size was set to 2 i.e. 1600 µs based on a MSO feedback. More MAP means messages mean more opportunities to transmit in the upstream. There is a tradeoff between choosing lower MAP-size with downstream bandwidth. Lower the MAP-size, more the downstream bandwidth consumed by Maps.

"max-round-trip-delay" is the RF RTT from a cable plant that can be configured in the CMTS. This delay should be adjusted based on expected distance between the Upstream burst receiver and the CM. In our experiments the value was set to 800 microseconds in propagation delay that equals to 100 miles in distance between the CMTS and CM.

Databackoff configuration can assist in spreading the effect of collisions in the broadcast request opportunities in a highly bursty and congested upstream. In this experimental setup we set it to a value in the range 5-8.

Figure 5 shows a high-level network diagram of a complex test rig to measure end-to-end DOCSIS Upstream latency. To emulate a typical MSO service tier, the CM's SLA was set to 100 Mbps in the DS and 20 Mbps in the US.

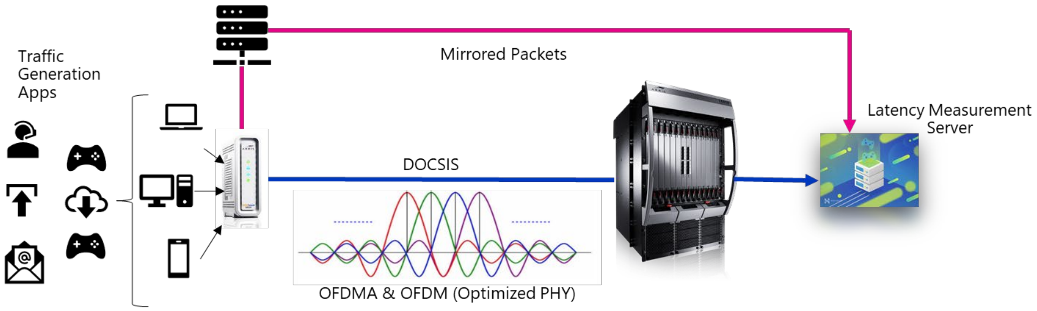


Figure 5 Experimental Test Setup

For all the experiments that were done using the PGS scheduling, the parameters were set as GGR = 2 Mbps, GGI = 1080 µs, GRI = 540 µs. The Weighted Round Robin scheduler is set to default weight of 9:1 ratio for the constituent service flows. The IAQM algorithm in LL SF is using default parameters of Ramp Function Exponent = 19 and the threshold = 1000 µs. The DOCSIS PIE AQM algorithm in CL SF is using a target latency = 25 ms based on MULPI spec recommendation to set AQM latency target between the range of 10 ms to 100 ms. The service flow buffer size is set to 50 ms based on previous experimentation and recommendations.

4. Experimental Analysis Of Latency In DOCSIS 3.1 System

Since the LLD architecture is defined in the realm of the DOCSIS 3.1 standard, we will first dive into an experimental analysis of latency in DOCSIS 3.1 system. CableLabs has used simulation studies to analyze the dual-queue architecture. While those studies were useful, real-world empirical studies using real-world non-deterministic data traffic model, real-world CCAP with real-world Schedulers and Mappers will provide even more valuable information to MSOs as they try to deploy LLD in the field.

Before we start our deep dive in to the experiments, it is important to note that all the information is experimental since a lot of MULPI spec specific development is still under development on the CMTS and CM software.

4.1. Single Upstream Service Flow Experiments

The first set of experiments used a DOCSIS 3.1 modem with AQM enabled on a single Service Flow emulating a single home with multiple users. This a traditional Service Flow setup that ingests all types of traffic i.e. QB and NQB. The AQM in this scenario is set to a target latency of 25 ms and uses DOCSIS PIE algorithm. In this experiment, we monitored the gaming traffic latency. One gaming stream was always transmitted in the LL SF and another in CL SF, in the case of LLD ASF experiments in order to measure the impact of QB and NQB on the gaming traffic.

Scenario 1

This scenario acts as a baseline experiment. The traffic mix included 2 UDP gaming streams in the US and DS directions, and a simple web browsing session.

Scenario 2

In this scenario, the traffic mix included 2 UDP gaming streams in the US and DS directions, 2 web browsing sessions, 2 ABR video streams (DASH) sessions that emulate OTT content like Netflix or YouTube, and an Upload speed test.

Scenario 3

In this scenario, the traffic mix included 2 UDP gaming streams in the US and DS directions, 2 web browsing sessions, 2 ABR video streams (DASH) sessions that emulate OTT content, and a file upload session emulating picture or short video upload.

Scenario 4

In this scenario, the traffic mix included 2 UDP gaming streams in the US and DS directions, 2 web browsing sessions, 2 ABR video streams (DASH) sessions that emulate OTT content, and a file upload session emulating picture or short video upload on a social media platform, and two video conferencing sessions.

Table 1 provides a summary of number of streams of different traffic mixes for all of the four Scenarios.

Table 1 Summary Of Number Of Traffic Pattern Streams Per Scenario

	🎮	🎮	💻	🎬	⏲	⬆	🎧
Scenario 1	**1x**	**1x**	**1x**				
Scenario 2	**1x**	**1x**	**2x**	**2x**	**1x**		
Scenario 3	**1x**	**1x**	**2x**	**2x**		**1x**	
Scenario 4	**1x**	**1x**	**2x**	**2x**		**1x**	**2x**

Table 2 covers all 4 scenarios that are mentioned above for single service flow setup.

The first scenario helped baseline the behavior of BE vs PGS for low-bit rate gaming traffic. The PGS enabled service flow showed a mean latency of ~1.5 ms and jitter of ~0.3 ms for the gaming traffic stream. Meanwhile, a BE enabled service flow showed a mean latency of ~5.5 ms and jitter of ~0.8 ms for the gaming traffic. This baseline behavior shows that use of PGS reduced latency by ~72%!

We can see that a single US SF with BE scheduling type has high latency (mean, 95th percentile, and 99th percentile) as the traffic mixes are increased in different scenarios. It is important to note that jitter* in all tests except Scenario 1 is pretty high. If there is anything that impacts a gamers QoE more than latency, it is jitter.

In all the experiments of this paper, Jitter is measured as the Mean Absolute Packet Delay Variation.

Table 2 Single Service Flow Gaming Traffic Latency With Best Effort and Proactive Grant Scheduling

Scenario	**Scheduling Type**	**Mean Latency (ms)**	**95 Percentile Latency (ms)**	**99 Percentile Latency (ms)**	**Jitter* (ms)**	**US Throughput (Mbps)**	**DS Throughput (Mbps)**	**PGS Efficiency (%)**
1 (Baseline)	BE	5.5	7.7	7.9	0.8	2	2	N/A
1 (Baseline)	PGS	1.5	1.7	2.7	0.3	2	2	44
2	BE	27	37	42	79	20	35	N/A
3	BE	27	43	50	142	20	35	N/A
4	BE	27	47	54	148	20	36	N/A

*Mean Absolute Packet Delay Variation

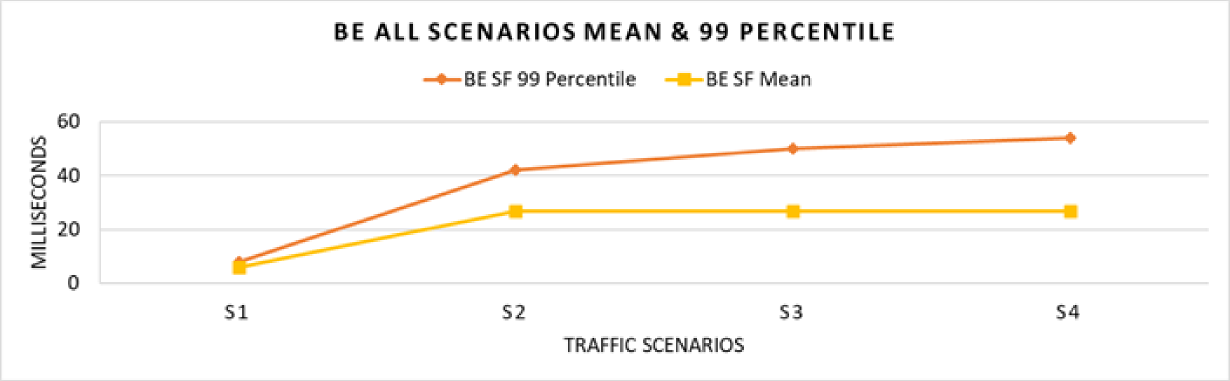


Figure 6 Single SF BE Mean and 99 Percentile Of Gaming Traffic

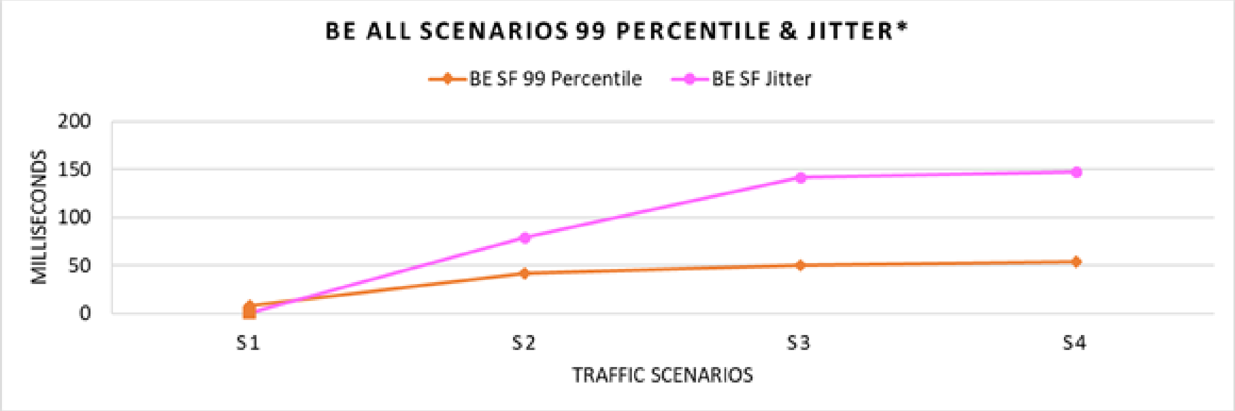


Figure 7 Single SF BE 99 Percentile and Jitter Of Gaming Traffic

4.1.1. Comparing Single Service Flow With LLD ASF Classic Service Flow

So how does the Single Service Flow latencies compare to the LLD ASF Classic SF latencies? In a Single Service Flow, QB and NQB traffic fill the SF buffers together, while in a LLD ASF there is a separation of QB and NQB traffic. We want to compare the latency metrics of gaming traffic in a Single Service Flow versus the latency metrics of gaming traffic within the Classic Service Flow that carries the QB traffic. The expectation in this experiment is that the latency metrics of gaming stream in a Single Service Flow will be approximately equal to gaming stream latency metrics within the Classic Service Flow.

Based on the data in Table 2 and Figure 8, it can be observed that the gaming traffic mean latency in CL SF is under the Single SF Mean up to Scenario 3, but for Scenario 4 the values are at a close approximation. These data points confirm that the theoretical expectation is correct.

The 99 percentile level for CL and single SF is approximately the same. This should be a good indicator that the NQB traffic within QB traffic in LLD architecture is fairly treated like the current circumstances of using a single service flow. In other words, latency performance isn't a zero-sum game. By separating

QB and NQB traffic, the NQB traffic can achieve much better latency performance, without degrading the performance of the QB traffic.

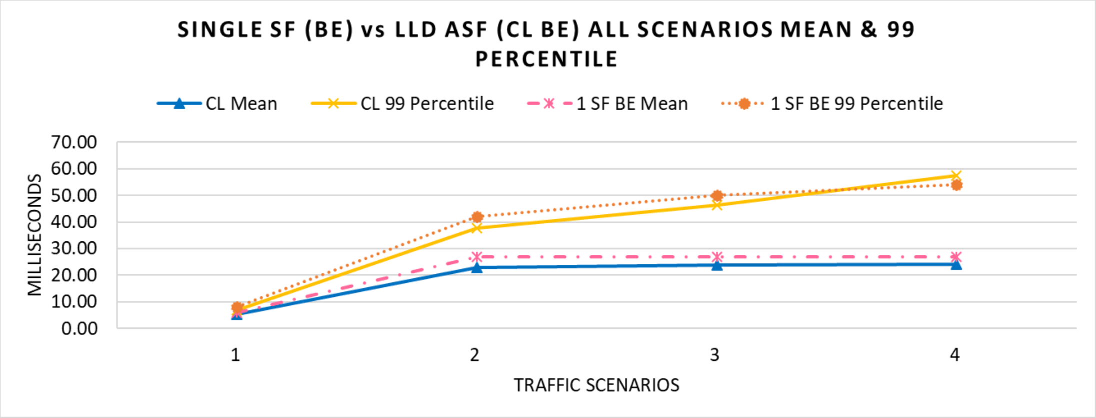


Figure 8 Single SF vs LLD ASF CL SF Mean And 99 Percentile Of Gaming Traffic

Figure 9 shows the 99 Percentile and Jitter data for the same comparison, and it can be observed that the CL Jitter is tracking to the single SF Jitter.

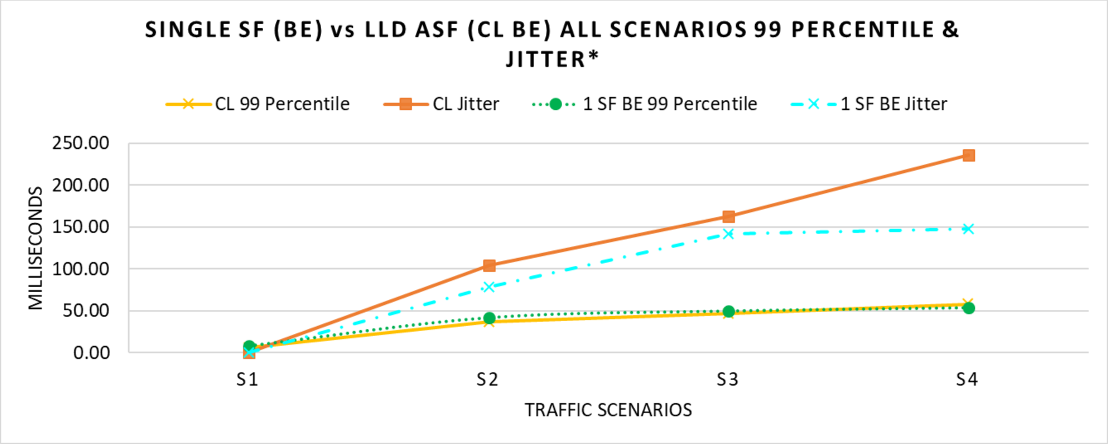


Figure 9 Single SF vs LLD ASF CL SF 99 Percentile and Jitter Of Gaming Traffic

4.2. LLD ASF With Best Effort Scheduling Experiments

This next set of experiments were focused on using LLD ASF, where a separation of Queue Building traffic and Non-Queue Building traffic was done by using relevant classifiers. By default, Queue Protection was enabled in these experiments and so was AQM as described in the experimental setup section.

The goal of this section is to highlight the importance of separating QB and NQB traffic. All the experiments were done with two configurations – 1. LL SF with BE scheduling and 2. LL SF with PGS scheduling. This is to contrast the behavior of two scheduling types.

4.2.1. Scenario 1

Table 3 shows the results of Scenario 1. The first scenario acts as our baseline for the separation of QB and NQB traffic since it includes only two gaming streams, one in each flow and the web session traffic in CL SF. It was observed that gaming latency is the same when the Best Effort scheduling is used in both the constituent SFs, CL and LL SF. Gaming traffic latency in LL SF is reduced by ~72% when PGS scheduling is used in the LL SF. The jitter also drops by ~70% when PGS is used in the LL SF.

Table 3 LLD ASF Scenario 1 Results

ASF With BE In LL SF and CL SF									
Scenario	Flow Type	Scheduling Type	Mean Latency (ms)	95 Percentile Latency (ms)	99 Percentile Latency (ms)	Jitter (ms)	US Throughput (Mbps)	DS Throughput (Mbps)	PGS Efficiency (%)
1.00	Low Latency	BE	5.33	6.81	6.97	0.93	1.00	3.00	N/A
1.00	Classic	BE	5.35	6.77	6.97	0.93	1.00		N/A
ASF With PGS in LL SF and BE in CL SF									
1.00	Low Latency	PGS	1.27	2.28	2.50	0.29	1.00	3.00	0.44
1.00	Classic	BE	5.24	6.68	6.89	0.86	1.00		N/A

4.2.2. Scenario 2

In Scenario 2, the traffic is increased significantly. The impact of QB and NQB is much clearer for either of the tests – with BE in both SF vs. with PGS in LL & BE in CL SF. The gaming traffic in LL SF continues to experience low latency and jitter compared to QB and NQB traffic mix in the CL SF.

Table 4 LLD ASF Scenario 2 Results

ASF With BE In LL SF and CL SF									
Scenario	Flow Type	Scheduling Type	Mean Latency (ms)	95 Percentile Latency (ms)	99 Percentile Latency (ms)	Jitter (ms)	US Throughput (Mbps)	DS Throughput (Mbps)	PGS Efficiency (%)
2.00	Low Latency	BE	5.54	7.13	7.37	0.88	1.00	35.00	N/A
2.00	Classic	BE	22.93	35.09	37.71	124.29	19.00		N/A
ASF With PGS in LL SF and BE in CL SF									
2.00	Low Latency	PGS	1.19	1.62	1.79	0.10	1.00	35.00	44.00
2.00	Classic	BE	23.41	33.51	37.29	104.34	19.00		N/A

4.2.3. Scenario 3

In Scenario 3, the traffic is more bursty because of frequent FTP file upload sessions. The Jitter experienced in CL SF is upwards of 150 ms which is detrimental to gamers QoE. The queues are filling faster, and the 99 percentile traffic waits longer in the queues even though AQM is trying to maintain the average latency of 25 ms. The gaming traffic in LL SF continues to experience low latency and jitter compared to QB and NQB traffic mix in the CL SF.

Table 5 LLD ASF Scenario 3 Results

ASF With BE In LL SF and CL SF									
Scenario	Flow Type	Scheduling Type	Mean Latency (ms)	95 Percentile Latency (ms)	99 Percentile Latency (ms)	Jitter (ms)	US Throughput (Mbps)	DS Throughput (Mbps)	PGS Efficiency (%)
3.00	Low Latency	BE	5.53	7.17	7.28	0.88	1.00	35.00	N/A
3.00	Classic	BE	23.73	40.74	46.21	143.82	19.00		N/A
ASF With PGS in LL SF and BE in CL SF									
3.00	Low Latency	PGS	1.23	1.61	1.74	0.10	1.00	35.00	44.00
3.00	Classic	BE	23.49	42.38	47.46	162.47	19.00		N/A

4.2.4. Scenario 4

In Scenario 4, the traffic adds to burstiness by adding a couple of video conferencing sessions along with frequent FTP file upload sessions. Note that the video conferencing data is passing through the CL SF. The Jitter experienced in CL SF is upwards of 200 ms which is detrimental to gamers QoE and also the video conferencing QoE. The queues are filling to latencies of 50 ms and more. The 99 percentile traffic waits longer in the queues even though AQM is trying to maintain the average target latency of 25 ms. The gaming traffic in LL SF continues to experience low latency and jitter compared to QB and NQB traffic mix in the CL SF, which is the same as the results found in the previous three scenarios.

Table 6 LLD ASF Scenario 4 Results

ASF With BE In LL SF and CL SF									
Scenario	Flow Type	Scheduling Type	Mean Latency (ms)	95 Percentile Latency (ms)	99 Percentile Latency (ms)	Jitter (ms)	US Throughput (Mbps)	DS Throughput (Mbps)	PGS Efficiency (%)
4.00	Low Latency	BE	5.58	7.02	7.31	0.88	1.00	39.00	N/A
4.00	Classic	BE	24.01	46.90	57.58	231.19	19.00		N/A
ASF With PGS in LL SF and BE in CL SF									
4.00	Low Latency	PGS	1.16	1.66	1.76	0.10	1.00	41.00	0.44
4.00	Classic	BE	24.27	49.07	58.02	236.24	19.00		N/A

4.2.5. Summary Of LLD ASF Experiments

The following graphs summarize the data in the tables for the four LLD ASF experimental scenarios.

Figures 10 to 13 signify the importance of separating QB and NQB traffic. Scenario 1 acts as the baseline and incremental traffic types are added up to Scenario 4. In general, it can be observed that the 99 Percentile latency of the traffic and the Mean latency and Jitter of the traffic in CL SF is >> than LL SF.

An MSO can choose to use Best Effort as the scheduling type in the Low Latency Service Flow and the Classic Service Flow. The applications that require high throughputs and which are not sensitive to latency can be classified into the CL SF, and they will be treated in a fashion similar to today's latency standards. On the other hand, the applications that are sensitive to latency can be classified into the LL SF, and they will be rewarded with latencies that are much better than today's latency standards. Furthermore, the additional enablement of PGS scheduling within the Low Latency Service Flow will reduce the latency metrics to millisecond values. But the PGS parameters must be carefully optimized so as to not overgrant or undergrant or a learning mechanism can be introduced to optimize the PGS grant efficiency.

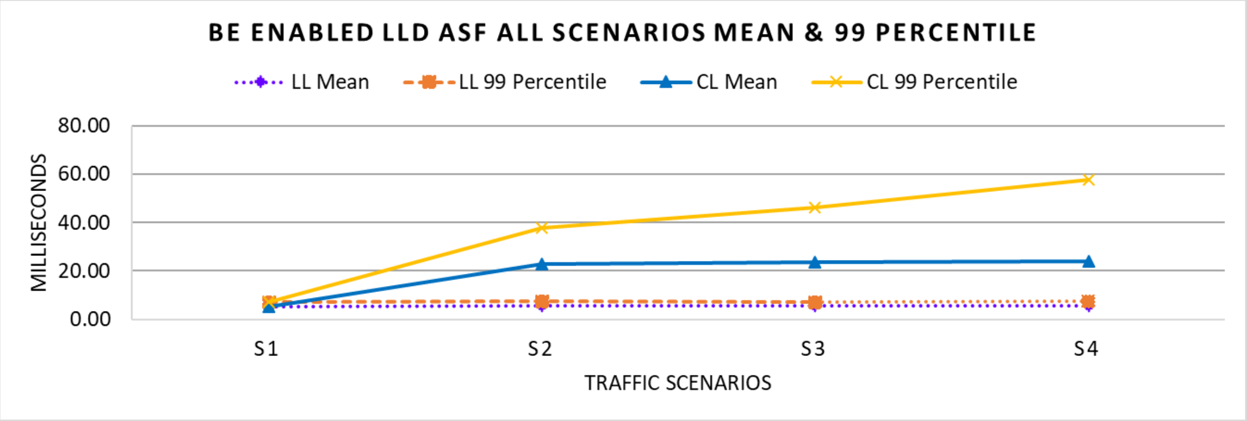


Figure 10 LLD ASF Both SF BE Mean And 99 Percentile Of Gaming Traffic

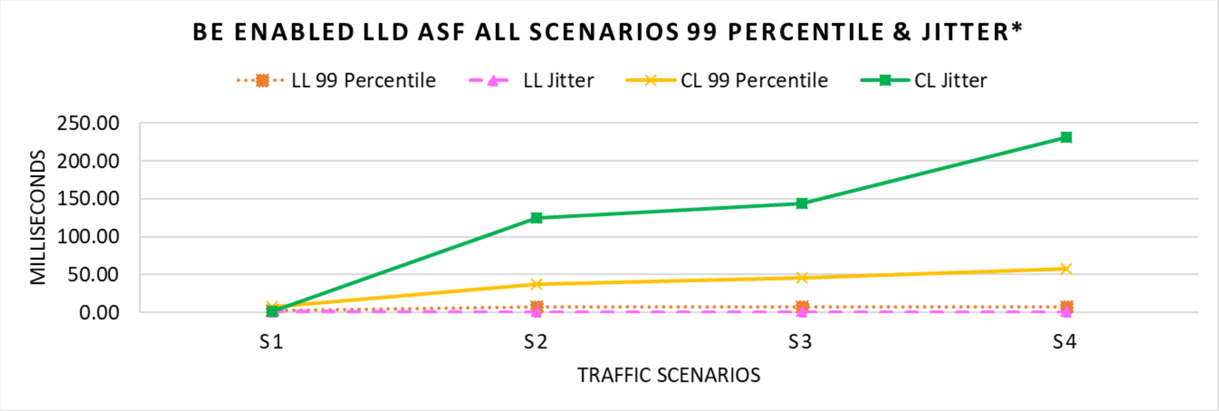


Figure 11 LLD ASF Both SF BE 99 Percentile And Jitter Of Gaming Traffic

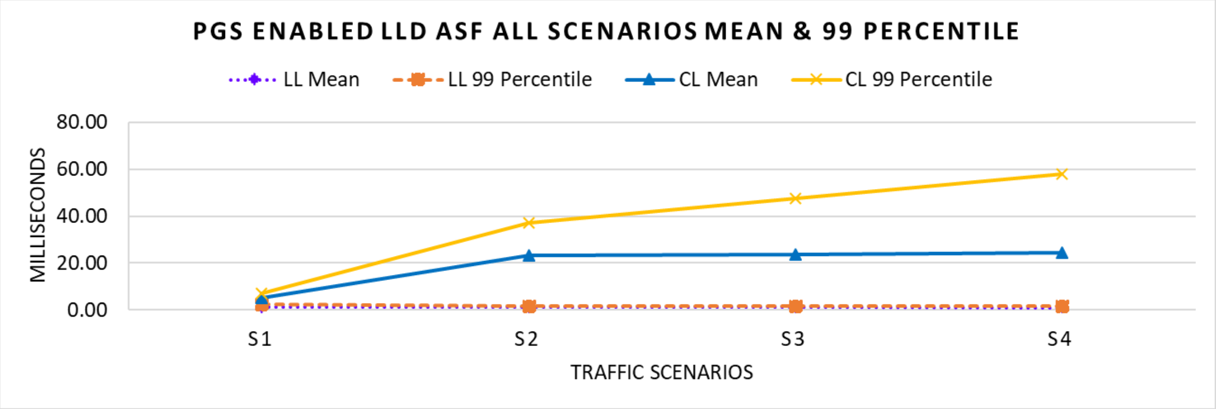


Figure 12 LLD ASF LL SF=PGS And CL SF=BE Mean And 99 Percentile Of Gaming Traffic

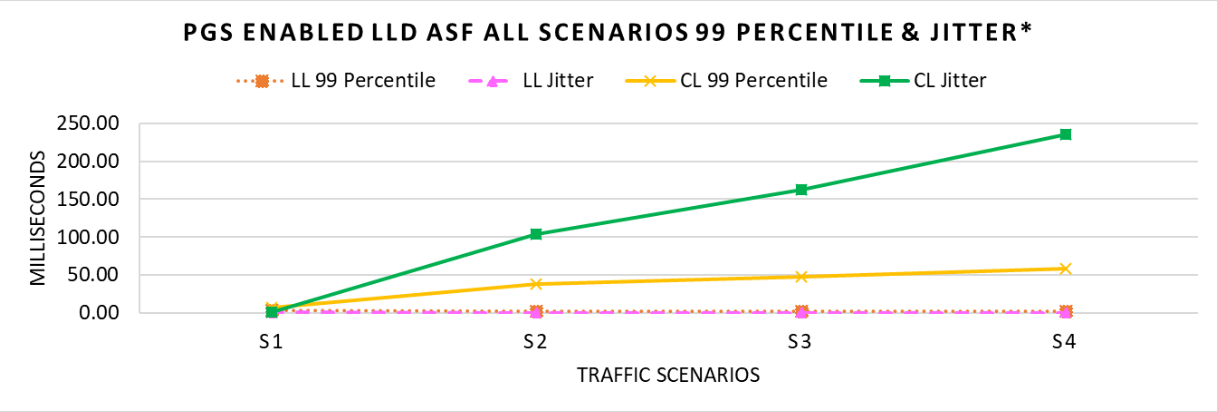


Figure 13 LLD ASF LL SF=PGS And CL SF=BE 99 Percentile And Jitter Of Gaming Traffic

5. Conclusions And Future Work

We proved that the concept of separating QB and NQB traffic assists in NQB traffic achieve lower latency than QB traffic, without degrading the performance of the QB traffic. We conducted experiments with the Proactive Grant Service Scheduling that send guaranteed grants at guaranteed intervals in order to meet the demands of traffic in the LL SF. We observed ~1.5 ms latency and extremely low jitter of 0.10 milliseconds for PGS enabled Low Latency SF within an LLD ASF.

The experiments also showed that either of the scheduling types can be used for LLD ASF – Best Effort or Proactive Grant Service. But it is important to note that Proactive Grant Service can reduce the latency further by proactively granting and reducing the traditional REQ-GRANT time by ~72%!

The CMTS and CM software and the MULPI specification are maturing to achieve the LLD goals. At the time of conducting the study, there were many moving parts because of many different features of the LLD architecture – Queue Policing, Weighted Round Robin, AQM algorithms etc.

There is a need for standard tools to measure latency statistics such as mean, 95 Percentile, 99 Percentile latency and Jitter values. LLD provides a Histogram feature that will give a deeper look in to the service flow buffers and queue build-ups. At the time of writing this paper, we did not have the standard toolset.

In the future, we plan to learn more about tuning of Proactive Grant Service parameters for optimized use of available upstream channel capacity and service flow Maximum Sustained Rate.

Since traffic classification into QB and NQB plays an important role, a future challenge lies ahead for the adoption of traffic classification rules in the LL and CL service flow at the application layer.

This real-world experiment with the CMTS and the LLD capable CM and the bursty traffic generators shows that low bit-rate traffic patterns (i.e. gaming traffic) that behave as NQB can achieve low latency with LLD ASF without impacting QB traffic.

LLD Interops at CableLabs are underway to ensure that the CMTS and the CM equipment are ready for deploying the LLD feature set. Once the spec and software are fully developed and tested, the introduction of LLD to the consumers will be exciting. LLD is one of the key enablers for the future of DOCSIS in the world of 10G.

Abbreviations

AMSR	Aggregate Maximum Sustained Rate
ASF	Aggregate Service Flow
BE	Best Effort Scheduling
bps	bits per second
CL	Classic SF
CM	Cable Modem
CMTS	Cable Modem Termination System
DSCP	Differentiated Services Code Point
ECN	Explicit Congestion Notification
FEC	forward error correction
HFC	hybrid fiber-coax
Hz	hertz
ISBE	International Society of Broadband Experts
LLD	Low Latency DOCSIS
LL	Low Latency SF
MSR	Maximum Sustained Rate
NQB	Non-Queue Building traffic
PGS	Proactive Grant Service Scheduling
QP	Queue Protection
QoE	Quality of Experience
QB	Queue Building traffic
SCTE	Society of Cable Telecommunications Engineers
SF	Service Flow
SLA	Service Level Agreement
WRR	Weighted Round Robin

Bibliography & References

[1] CableLabs, Low Latency DOCSIS: Technology Overview, G. White, K. Sundaresan, B. Briscoe

[2] MAC and Upper Layer Protocols Interface Specification, CM-SP-MULPIv3.1-I20-200407, April 04, 2020, Cable Television Laboratories, Inc.