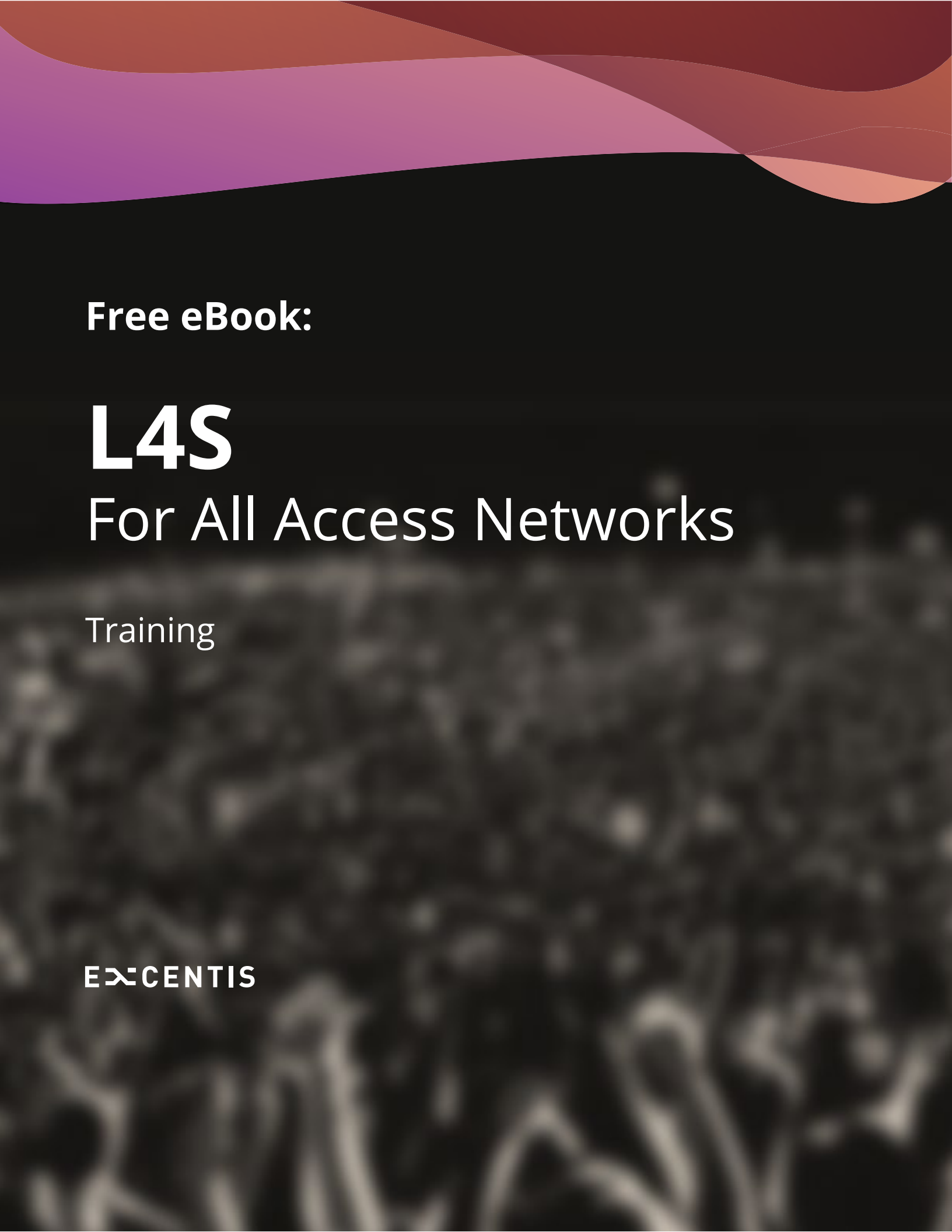
Free eBook:

L4S
For All Access Networks

Training



EXCENTIS

Overview

Internet access speeds have long been the primary factor for comparing various services. However, the emergence and widespread use of new applications have shifted this paradigm. While speed remains crucial for uploading or downloading files, latency has gained equal importance, especially for real-time applications such as videoconferencing, gaming, etc.

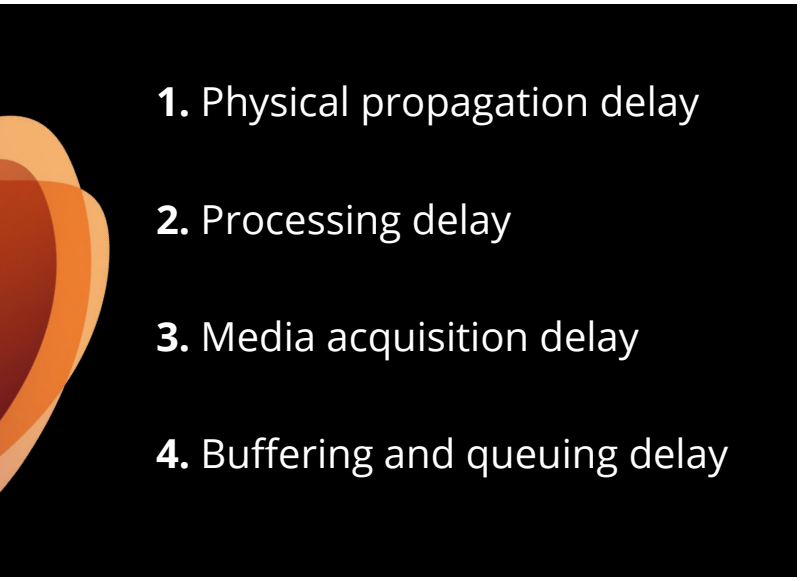
Research indicates that while higher speeds marginally increase the Quality of Experience (QoE) for users, the decisive factor becomes latency.

It's essential to note that packet loss, a key factor known since the inception of VoIP, remains crucial. However, while latencies around 100 ms were once acceptable for VoIP applications, the array of new applications demands latencies below 10 or even 1 ms.

Table of Contents

Latency Contributors on the Internet

When considering methods to reduce internet latency, it is essential to understand the various sources contributing to it. Without delving too deeply into each one, they can be categorized as follows:



Let's briefly have a look at each of those:

Physical propagation delay

We're all aware that nothing surpasses the speed of light in a vacuum (please refer to Einstein's paper at https://www.fourmilab.ch/etexts/einstein/specrel/specrel.pdf if you're uncertain about that). Thus, no matter the efforts made, when transmitting a signal over 3000 km, it will take at least 10 ms to reach the other side. Unless we develop some Star-Trek-based technology, there's nothing we can do about that.

Processing delay

Processing delay comprises primarily two components. One component involves the time required to examine an incoming packet on an interface and determine which interface to forward it to. Thus, any intermediary device (such as a switch, router, firewall, etc.) requires a certain amount of time to inspect the packet. The additional latency will be contingent upon the resources available on the device and its current workload.
The second type of processing delay is induced by the specific link technology in use. Error correction techniques and time interleaving serve as measures to counteract certain types of noise. The extent of added latency varies significantly depending on the technology employed. While it is easy to calculate, the amount can fluctuate based on the size of the transmitted packet.

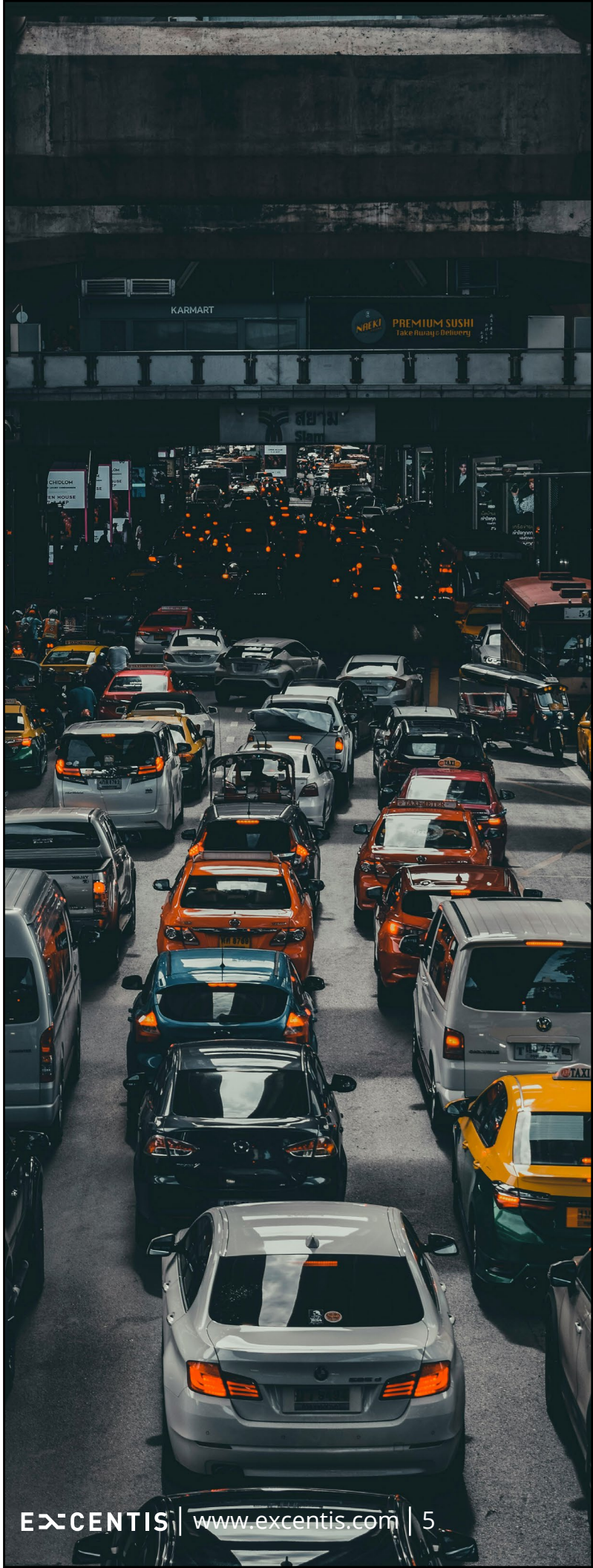
Media acquisition delay

This type of latency occurs in systems where multiple transmitter/receivers share bandwidth or resources on the medium. Wi-Fi provides a notable example where media acquisition represents the primary source of latency in specific cases. In a simplified example: if a device operating at a low speed (due to its distance from the access point) transmits a large packet, this transmission will take some time (on the order of milliseconds). During this period, no other device will be able to transmit, leading to media acquisition delay for other devices on that Wi-Fi network. When multiple devices attempt to transmit simultaneously, the resulting media acquisition time can become quite extensive.

Buffering and queuing delay

Buffering and queuing delay occurs when an interface cannot deliver the required speed based on the number of bytes/packets it must transmit. This can happen due to media acquisition or because input from a higher-speed interface needs to be sent through a lower-speed interface, or when traffic from multiple interfaces needs to be transmitted via a single output interface. In such cases, switches, routers, etc. will buffer packets to prevent packet loss. It's important to note that buffers have a limit in size, so once they are full, packet loss becomes inevitable. While larger buffers decrease the chance of packet loss, they also extend the maximum time a packet stays in the buffer, thereby increasing maximum observed latency. This observation is fundamental.

For understanding the rest of this e-book: buffers are beneficial for minimizing packet loss, but excessively large buffers are detrimental as they amplify latency. Regrettably, there's no single optimal configuration for today's internet traffic since it hinges on the requirements of your applications.



TCP – The Transmission Control Protocol

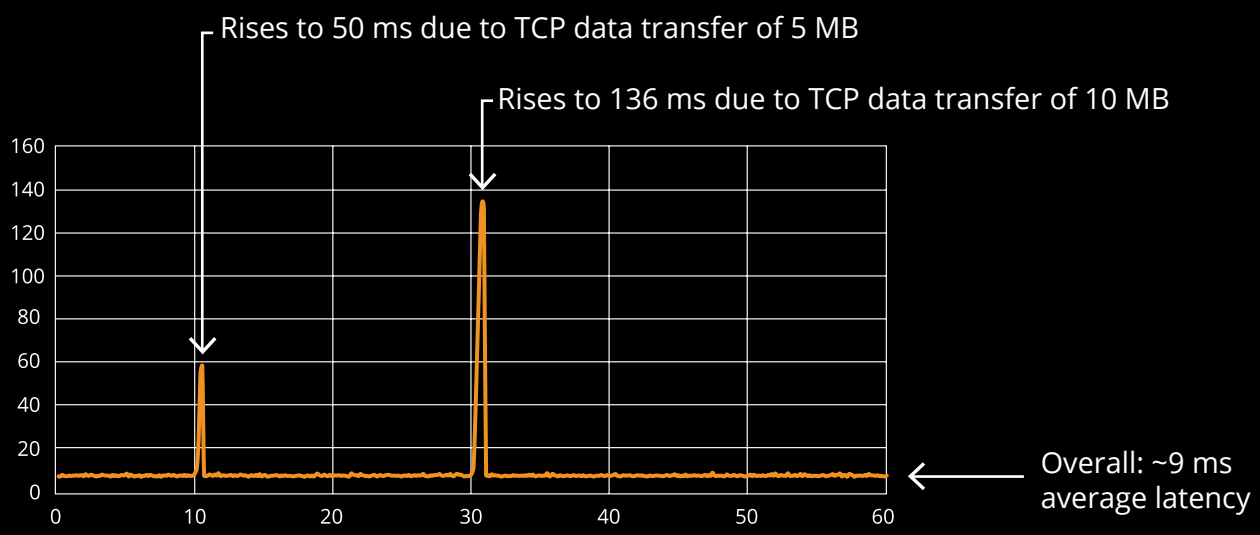
Research* has shown that 90% of internet traffic is TCP. For an exhaustive explanation of current TCP standards, interested individuals can refer to the relevant RFC documents accessible on the IETF website. The primary characteristic is that the TCP-algorithms (such as Reno, Cubic, etc.) are congestion-seeking algorithms. In simpler terms, a sender continues to increase its transmission speed until congestion is detected through packet loss. This detection of packet loss, disregarding other sources for this discussion, indicates that the buffers are full, thereby introducing a certain level of latency (depending on the buffer size).

Consider a scenario where a customer is doing a real-time application, sending 64 packets per second (a typical figure for real-time action games) while concurrently uploading a file (using TCP) of a certain relatively large size (e.g. 5 MB). Before the upload begins, the latency hovers around 9 ms, providing the customer with a satisfactory gaming experience. However, once the upload starts, the gaming packets will end up in the same buffer as the TCP packets and the latency significantly increases.

As depicted in the graph, a file transfer of 5 MB makes the latency on gaming traffic to rise to 50 ms. If the file transfer increases to 10 MB, the latency surges to 136 ms, enough to crash your car because you saw your opponent make that manoeuvre too late.

Research by Impact of Evolving Protocols and COVID-19 on Internet Traffic Shares.
Luca Schumann, Trinh Viet Doan, Tanya Shreedhar†, Ricky Mok, and Vaibhav Bajpai. TUM, Germany, IIT-Delhi, India, CAIDA, USA.

Average Latency [ms] - UDP flow 64 pps



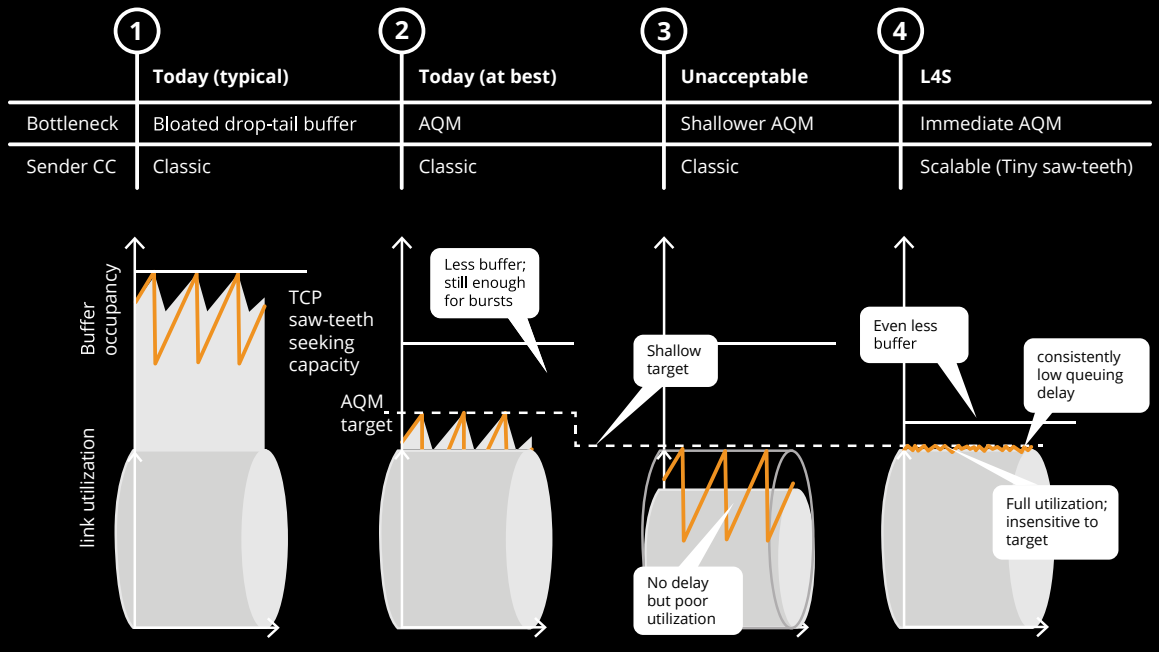
L4S: Unveiling a Transformative Paradigm

The basics of L4S – an architecture for Low Latency, Low Loss and Scalable Throughput (specified in RFC9330) are rooted in the concept that the transmitters, rather than the queue itself, are the primary cause of delay. In other words, the transmitter operates too quickly, causing congestion and resulting in additional delay. Therefore, the solution is slowing down the transmitter before the buffer becomes a significant source of latency.

The fundamental idea is to signal back to the transmitting side that latency is increasing and prompt the transmitter to adjust its rate according to the current level of latency (buffer occupancy).

The advantages of signalling congestion clearly by providing feedback before packet drops occur is nicely illustrated in the figure below.

In today's traditional TCP (see image No.1) (before AQM), the buffers consistently remain quite full, resulting in substantial delays. It does have the advantage of fully utilizing links. The second image illustrates the utilization of Active Queue Management (AQM). (see image No.2) Once the buffers reach a specific occupancy threshold, random packet drops are introduced to slow down TCP and reduce latency (as the buffers are less occupied). However, the challenge lies in selecting the appropriate configuration parameters for buffer occupancy (AQM target) . If its chosen too small, the latency is indeed minimal, but the link is underutilized (see image No.3). L4S, as shown in image No.4, using immediate AQM, consistently maintains the buffer at low occupancy while fully utilizing the link. (see image No.4) So, it achieves both low latency and high link utilization!



Source: Implementing the 'Prague Requirements' for Low Latency Loss Scalable Throughput L4S. Bob Briscoe, Koen De Schepper, Olivier Tilmans, Mirja Kuhlewind, Joakim Misund, Olga Albisser, Asad Sajjad Ahmed.

The details of the L4S algorithms are beyond the scope of this ebook and are fully explained in RFC9331. The importance lies in the fact that the sender transmits at a rate that maintains nearly full link utilization while also keeping the buffer occupancy minimal to minimize latency.

When congestion is detected through packet loss, and for the sake of this discussion, other sources of packet loss are disregarded, it implies that the buffers are full. Consequently, a certain amount of latency (dependent on the buffer size) is introduced due to the full buffers.

L4S does however require some modifications on the currently installed internet infrastructure. First, IP-stacks on end-devices (laptops, servers, tablets, smartphones, etc.) will need updates to incorporate the L4S algorithm (congestion notification, transmission rate adaptation). Secondly intermediate devices (routers, switches, Wi-Fi APs, etc.) also need to integrate the L4S requirements because it is within these devices that queues are formed. These devices must detect the L4S packets and set the notification bits in the right packets if packets start to accumulate. That notification functionality is part of the Explicit Congestion Notification (ECN). L4S uses this mechanism.

Nevertheless, if any intermediate device performs ECN bleaching (this means that always overwrites ECN notification bits), it will disrupt the L4S transmission. It is a part of the L4S requirements that the software stacks should revert to legacy TCP operation in those cases.

It is important to note that intermediate devices that are not L4S-capable but that do not overwrite the L4S bits (they don't perform ECN bleaching) will not "disrupt" the L4S system. However, to achieve benefits from the L4S technology, the devices situated at the edges of congested links should implement the L4S requirements. Sending packets over a slow-speed access link without L4S capabilities and then traversing a L4S-capable backbone without any congestion will not yield any advantages because the access link will be the one dropping the packets. In such a scenario, no congestion marking will occur.

L4S: Synergizing with Legacy Infrastructure

Unfortunately, it is not possible to flip switch the internet to the new architecture. What is required is a pathway that ensures the seamless operation of legacy applications while coexisting harmoniously with the new applications utilizing L4S. To address this issue, researchers devised the Dual-Queue Coupled Active Queue Management (AQM) for L4S as outlined in RFC9332.

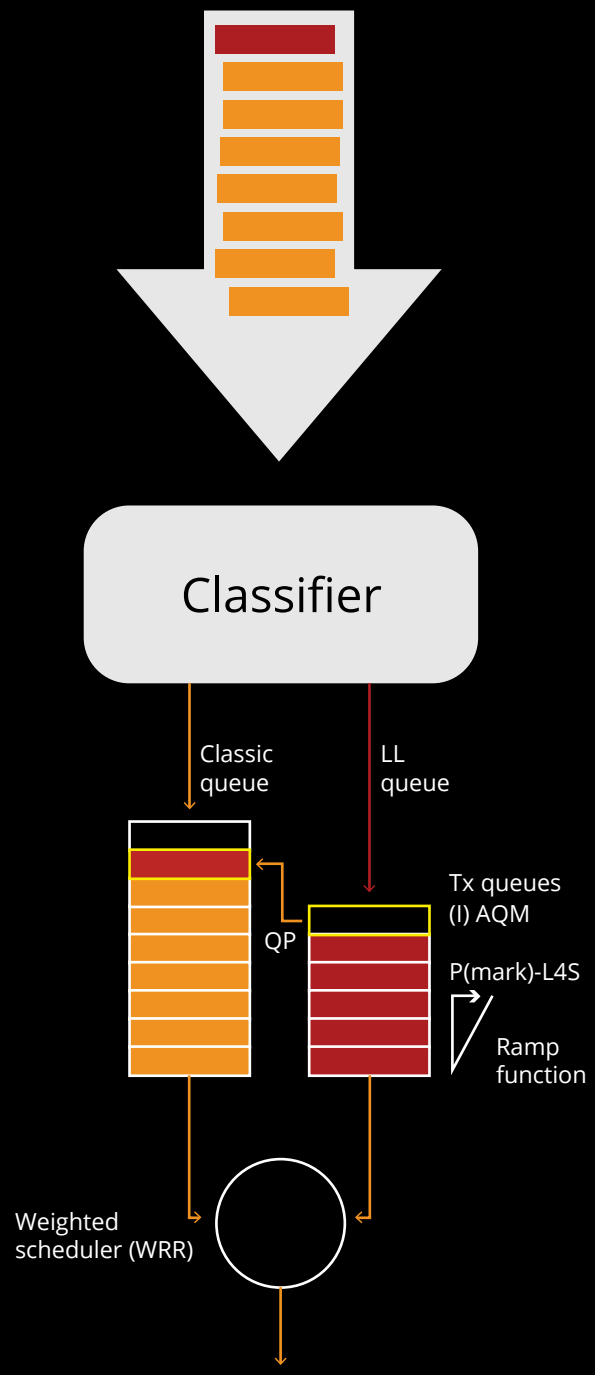
The simplified illustration of this mechanism is illustrated in the figure on the next page:

This mechanism should be implemented by the intermediate devices (such as routers, switches, etc.) that are L4S capable. Packets transmitted by L4S-capable stacks are identified and directed to the Low Latency (LL) queue based on classifiers. Other packets are routed to the classic queue. Packets out of the LL queue that are transmitted will have a probability to get marked with a congestion notification based on configuration parameters (e.g., max allowed latency) and the current latency introduced by the queue. If the queue is relatively full, congestion notifications will occur more frequently than when the queue is nearly empty. The latency threshold at which this marking can occur is also a configuration parameter. Before transmitting the packets on the interface, a weighted round robin scheduler selects packets out of the classic queue or the Low Latency queue. The weight should be chosen such that there is a fair treatment of both the L4S and classic traffic, allowing each to access their fair share of the available bandwidth.

As there is a risk of certain senders or applications misbehaving —specifically, those flows that do not adjust their rate during congestion —additional protection measures are necessary. DOCSIS® introduced a Queue Protection (QP) mechanism for that purpose that packets that belong to a flow that contributes most to the latency are re-classified to the classic queue. Other arrangements, such as dropping on saturation, can also be used, in such cases, the latency will never exceed the maximum defined target.

Dual-Queue Coupled Active Queue Management (AQM) for L4S as specified in RFC9332.

Simplified mechanism illustrated below



L4S: Unlocking the Benefits

To reap the benefits of L4S it is crucial for it to be implemented initially on the weakest links along the path between the sender and receiver. This is where congestion will occur, so this is where congestion can be indicated. Many times, this might be the access network, but often this will be the in-home Wi-Fi network. L4S has been evaluated and demonstrated in Wi-Fi networks, by researchers from the University of Edinburgh and the University of Glasgow, showing that L4S can achieve low latency and high throughput for both short and long flows. *[https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1484466&dswid=-2544]* It is however known that Wi-Fi can be an incredibly challenging and dynamic environment to operate in, especially in a multi client/ multi-SSID environments. Certainly, more research and experimentation will need to be conducted here. Right at this point is where a tool like ByteBlower, serving as both a traffic generator and analyser, can be of immense assistance!

> "L4S can achieve low latency and high throughput for both short and long flows."

L4S: Testing Devices and the Network

When testing L4S equipment, networks and its configuration, it is especially important to use realistic traffic patterns as much as possible. If the only flow on a network involves gaming at 64 packets/second, achieving low latency is not a significant challenge. However, the scenario where you are simultaneously playing an online game, downloading a software update on the gaming console, and a third user in the household engages in a video call is much more reflective of today's situations. The scenario represents the use case where customers are unhappy with their online experience.

Therefore, testing for L4S requires IP traffic generator/analysers that can create realistic traffic patterns and conduct valuable measurements that offer insight of the root causes of unexpected results. Moreover, it should function effectively in both wired and wireless (including mobile/5G) scenarios. This is where the ByteBlower becomes instrumental.

ByteBlower, the Excentis TCP/IP traffic generator, offers essential features for conducting test and analyses focused on latency, packet loss, and throughput – key parameters optimized by L4S:

ByteBlower Key Features

1. Accurate latency measurement over time, including average, maximum and minimum per time snapshot.

2. Distribution function of the latency for all packets so one can see with the blink of an eye if certain requirements on e.g. 99% percentiles are met.

3. Capability to mix classic (non-L4S) and L4S traffic using the same tool for both TCP and UDP flows.

4. Works on wired (Ethernet) interfaces, and mobile devices, facilitating tests across wired and non-wired access networks like mobile (5G,...) and Wi-Fi.

5. Support for multi-client tests enabling multiple devices to participate simultaneously. Each flow can be initiated and terminated at fully configurable times.

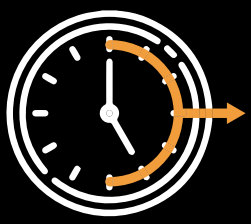
6. Comprehensive reporting on key TCP parameters such as round-trip times, packet retransmission, and congestion notifications.

Don’t let inefficient network testing processes hold your company back. Embrace the power of

ByteBlower

and unleash the true potential of your team.

Reduce time and effort for network testing.



ByteBlower’s **user friendly** interface and automation capabilities dramatically reduce the time and effort required for network testing.

Focus on what matters the most.

Reduce 50% off a team member’s workload

This could translate to savings up to

$40.000 to $50.000

a year.

Boost your engineering team’s productivity by halving the workload of one engineer, allowing them to concentrate on other essential tasks. *

Let’s advance networks together

Contact us →

*Reference table

	APAC	EMEA	NAMER	
Average wage cost	$80.000	€90.000	$100.000	per year
Savings with ByteBlower up to	$40.000	€45.000	$50.000	per year

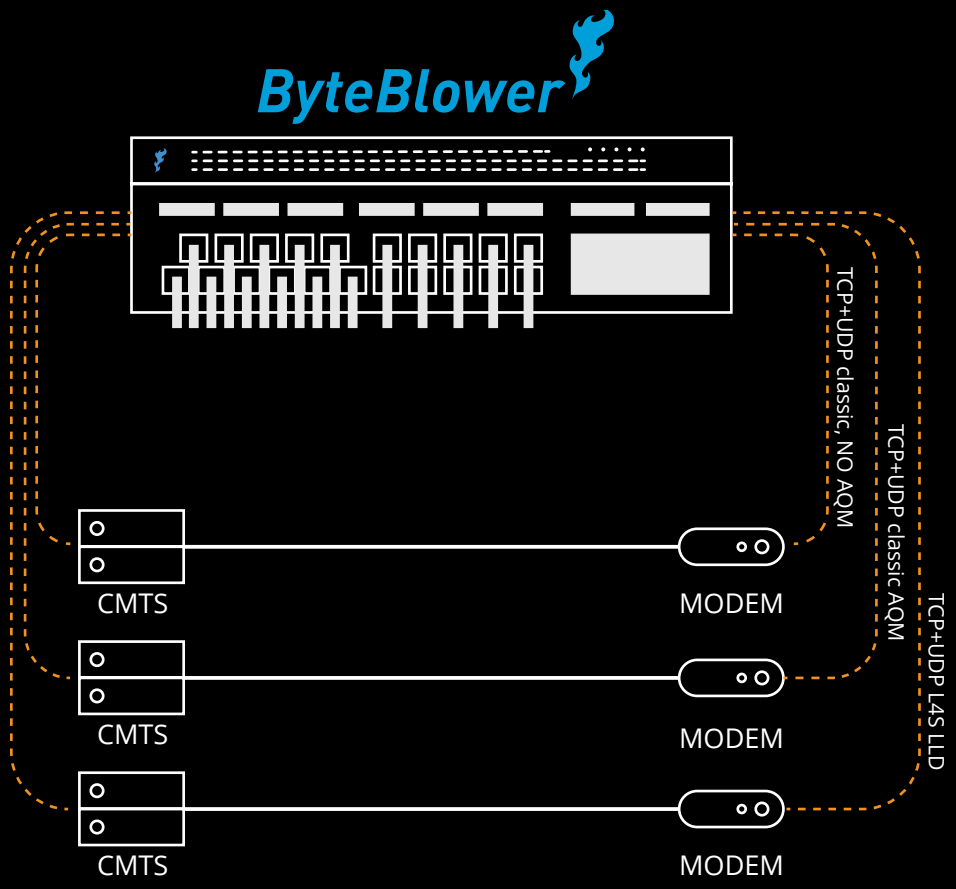
(Indeed, 2023)

L4S: Potential Achievements

L4S can significantly reduce latency while still maintaining high throughput and low packet loss, as illustrated by the results of the experiments conducted on a DOCSIS® system below. The tests involved three parallel setups using a single ByteBlower.

The objective of the test was to compare the observed latency on a UDP flow of 64 packets per second while simultaneously executing a TCP data transfer.

> "ByteBlower offers essential features for conducting test and analyses focused on latency, packet loss, and throughput."



In this experiment, the L4S latency target for the cable modem was set for 30 ms. Therefore, it can be observed that this target was met for 99.9% (the additional 3 ms can be attributed to other components of the system)

It can be observed that L4S meets its objectives of keeping the latency below the targeted value. In comparison to AQM (considered the current best), latency is reduced by over 100 ms in for the 99.9% reading, improving the quality of experience significantly.

It is important to highlight that whereas in the past, focus was primarily on the average latency across all packets in a flow, the latency distribution has now become the primary criterion, particularly the 99 or even 99.9%. Having an average latency of 40 ms but 1% of packets experiencing a latency of 500 ms equates, for real-time applications, to the impact of 1% packet loss. See results in the table below.

Results Experiment 1

% of packets having latency less than	Classic	AQM	L4S
90%	220 ms	20 ms	9 ms
99%	220 ms	23 ms	30 ms
99.9%	225 ms	140 ms	33 ms

In another experiment we measured the latency of a fixed-rate UDP flow while also performing a TCP download. The goal was to verify the impact of using L4s for both UDP and TCP.

In the drawing below, the left hand side represents the L4S results while the right hand side represent the classic results. For both L4S as classic AQM the latency target was configured for 10 ms.

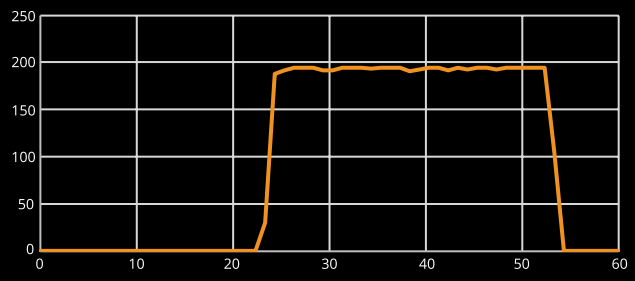
As you can see, the achieved TCP throughput is almost the same for both cases. The L4S TCP however has significantly less sawtooth behavior.

When looking at the UDP latency, one observes that with L4S the latency is kept below 10 ms, while the classic AQM mechanism only succeeds to keep it below 30 ms.

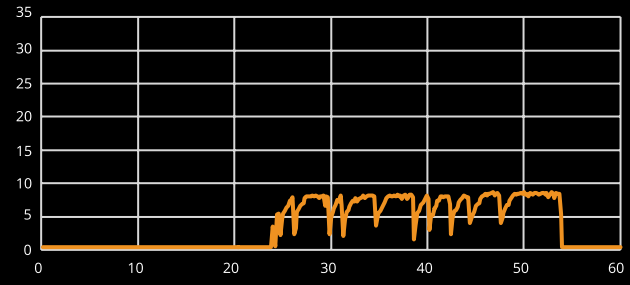
Results Experiment 2

L4S

TCP Throughput [MBit/s]

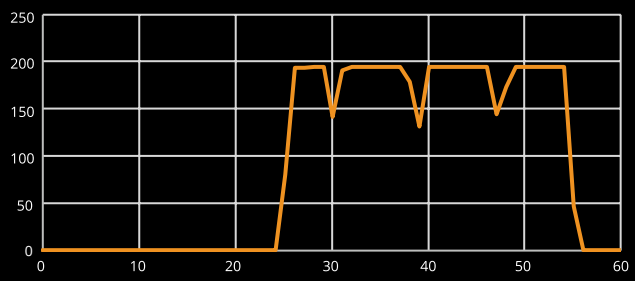


Latency [ms]

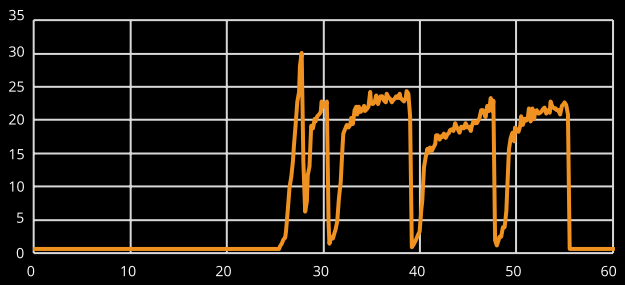


Classic

TCP Throughput [MBit/s]



Latency [ms]



Key Findings

With the advent of latency becoming a determining factor for user Quality of Experience, L4S is a key component in making the internet ready for the future set of applications. While it's still in the early phases, initial results are promising and show that latencies can be reduced significantly. Lab experiments and field trials will have to show the optimum set of parameters. Stay tuned on to keep up to date with results from the fields. If you are aware of devices that already implement this new technology, please let us know at L4S@excentis.com

It can be observed that L4S meets its objectives of keeping the latency below the targeted value. In comparison to AQM (considered the current best), latency is reduced by over 100 ms in specific scenarios, improving the quality of experience significantly.

It is important to highlight that whereas in the past, focus was primarily on the average latency across all packets in a flow, the latency distribution has now become the primary criterion, particularly the 99 or even 99.9%. Having an average latency of 40 ms but 1% of packets experiencing a latency of 500 ms equates to 1% packet loss. For real-time application, packets that arrive above a certain latency have the same impact as packet loss.

As we wrap up our exploration of L4S, remember that learning is a continuous journey. This eBook provides a comprehensive understanding of L4S technology, empowering you to transform network communication for faster, more reliable experiences worldwide. Stay connected for future updates and support as we navigate the evolving tech landscape together, advancing towards a more efficient digital world.

We're humbled to be trusted by the best.

Advancing the network of today, paving the network of tomorrow.

Let's advance networks together

We would welcome the opportunity to work with you to optimize, innovate and assure the robustness of your networks. And we put our heart into it, our work is our passion.

Jan de Beule
EMEA

Contact

Charlie Viaene
AMER/APAC

Contact

We're known for exceptional products and word-class independent expertise in testing and training services.

Focusing on the needs of our customers is the core of our business and this for more than two decades.

EXCENTIS
EXCELLENCE IN CONNECTIVITY
www.excentis.com

Excentis Europe
+32 9 269 22 91
Gildestraat 8
9000 Ghent
Belgium
info@excentis.com

Excentis USA
+1 347 720 6896
530 7th Avenue-Suite 902
New York, NY 10018
USA
info@excentis.com