Lower-layer performance is not indicative of upper-layer success

Lucas Pardue (Cloudflare), Sreeni Tellakula (Cloudflare)

RFC 8890 [1] explains why the IAB believes that the Internet is for end users - humans that perform activities supported by networking standards, technology, implementation, deployment, and all the things in between.

End users can be impacted by network quality but should not be expected to have empowerment to effect any change beyond the local domain. And even then, external factors might restrict the illusion of control. One might think that network actors such as Internet Service Providers are empowered to improve Quality of Experience (QoE) but the truth is they are restricted too.

This paper attempts to show the complexity of one small slice of end-to-end behaviour, and explain why extrapolating findings from slices of behaviors in order to effect change or decision making is unlikely to meet expectations.

Users are driven by goals

End users don't interact with the Internet for the sake of it. Most human use is direct and goal driven (e.g. sending email, receiving a video call, browsing the Web), and supported by applications. The relationship between application and Internet is an indirect and asymmetric dependency from the user perspective. Humans will have various levels of technical understanding of what role the Internet plays in achieving their goals. The lowest common denominator being that "any Internet connection" is required. This is, in a sense, similar to utilities like water and electricity.

The comparison of the Internet, water, and electricity is an interesting tangent. Water can be used for direct human consumption, cooking, cleaning, industry or a multitude of other things. The quality of water itself can be measured, affecting what it is allowed to be used for [2]. Electricity, in contrast, is not intended for direct human consumption [3]. In turn, the majority of the population without electrical qualifications will struggle to understand how the quality of an electrical signal [4] impacts their usage of it. The availability and reliability of both water and electricity are (mostly) orthogonal to its quality. When those degrade, there can be an impact on human users, either directly (they can't drink) or indirectly (they can't illuminate a room). Fallback measures typically involve out-of-band solutions (water barrels, uninterruptible power supplies, generators, etc) and rarely complete system failover.

Humans don't consume the Internet directly, their goals may require them to use it. The goal "I want to be entertained", could be satisfied by watching a movie via physical media or an Internet streaming service. A goal's quality of experience is often linked to the efficacy of the chosen solution. Where this depends on the Internet, what we're usually saying is that it depends on interacting *prime* systems. A local application, a remote application, and a network path (or paths) between them can all be considered prime — each in reality is a system of systems. For instance, applications have layers (User Interface, business logic, databinding) that drive the network interaction element [5]. The process of achieving an overarching goal, therefore, is a sequence of interactions between layers and systems. How well those interactions perform is key to providing quality.

What factors impact system interactions?

Network-minded people might be quick to suggest the network path is the most important factor in systems interactions. For example, larger files will likely traverse a network faster when there is more bandwidth available. But that doesn't directly translate to how well a user goal is achieved because there's subjectivity and subtly that is easily overlooked.

If a human is attempting to transfer a picture, the goal might be "Retrieve a medical scan for diagnosis in a timely manner", "Obtain any size picture of a dog, as fast as possible", or "Ensure this image is uploaded to backup storage, it is not important that it happens quickly". Throwing bandwidth at the problem is rational thought but might not improve the irrational human's subjective experience. Applications could optimize pictures (or other resources), for example with scaling or compression, in order for them to transfer more quickly over fixed bandwidth. Alternatively, there are many human perception tricks that can be employed to give the impression that goals are achieved before they really are.

The network doesn't always play a role in how humans use the Internet. Considering HTTP as an example, an oft-quoted line is "The best and fastest request is a request not made." [6]. Client-side caching can avoid the need for network interaction, which simplifies the steps and can help to achieve better performance. Interestingly though, local cache storage access times can actually turn out to be worse than network access times. Mozilla Firefox has a feature called RCWN (Race Cache with Network) that races the two access methods. The ticket [7] that enables this feature states "Telemetry from 2017/07/27 to 2017/07/31 shows that average time saved by racing was 5.82s while the median was 150ms.".

The factors that affect system interactions, and their performance, can affect the choice of system interactions. This is a *complex system*, and an analysis is beyond the scope of this paper. For our purposes we can assume that networks will always play *some* part in how humans use the Internet. And in those cases path bandwidth/throughput/goodput, latency, loss, and jitter are the well-known factors that affect the behaviour of the protocols being used by applications.

Shifting sands on shifting sands

There continues to be a rapid and ever growing variety of Internet-based applications that help humans achieve their goals. While Internet transport and application protocols have also evolved, they have not proliferated at the same rate. That's a testament to the flexibility of such protocols.

Evolution, however, is not completely unbounded. Network ossification has severely hampered transport protocol development. Today, we're left with TCP and UDP as the only widely-deployable protocols on the Internet. Even those two receive very different, conditional treatment, often based on the applications that have run atop them, for good or bad.

What we're seeing is the consolidation of application protocols onto TCP or UDP. And further, a growing trend in using HTTP as a substrate (see BCP 56 [8], and its soon-to-be replacement draft-ietf-httpbis-bcp56bis [9]). So it cannot be taken as given that these protocols, and the applications that drive them, will all continue to use the network path in the same way. Understanding exactly what network factors affect user-facing quality is complex.

When things are complex, humans tend to over-simplify. A class of measurement, so-called speed testing tools [10] [11], attempt to measure the performance characteristics of the path and present them to a user for interpretation. Larger is better, maybe. The applicability of these measurements is open to debate. End users are unlikely to be able to run such tests themselves against an Internet Service Provider they are not contracted with, preventing them from proactive due diligence tests. Information gathered from speed tests may be comparable to other recorded information, however users are unlikely to understand the accuracy or source of active or recorded measurement. Furthermore, users are not empowered with accessible information about the path elements composing the network test, nor are they likely to be empowered to change much about the elements that have been measured.

Measure for measure

Measurements of network performance say *something* about path characteristics and that is possibly better than *nothing*. However, they can often be contrived examples of how end users actually exercise the path. A measurement of network bandwidth that scores well (in whatever metric is being used) does not necessarily translate to good QoE for heterogeneous applications accessed over that path. This is true even when the transport and application protocol is common.

For example, different types of web page workloads can be more or less sensitive to latency or bandwidth, depending on the user's goal. As user expectations of the web have grown, sites and pages have grown in complexity, leading to user agents and web servers interactions with the network being as important as the network itself. Speed tests are exceedingly unlikely to exercise the network in the same way, emulating the diversity of the Web is difficult.

RFC 7594 [12] describes a framework for Large-Scale Measurement of Broadband Performance (LMAP), with one of the given deployment examples in section 6.4 being similar to a speed test setup. RFC 7536 [13] describes LMAP use cases whereby end-to-end QoE might be understood by linking network measurements to a service-dependent "mean opinion score". Mapping parameters to opinion scores is a complex matter. BCP 170 [14] touches on the matter, with some examples oriented to real-time applications such as RTP (RFC 7266 [15]) and video codecs (draft-ietf-netvc-testing [16]). How the network maps to mean opinion scores for other application protocols, such as HTTP, appears to be an open question. In contrast, there seems to be a variety of documents describing frameworks for TCP/IP oriented testing e.g. ITU-T Q.3960 [17], RFC 2330 [18], and RFC 6349 [19].

Looking more closely at HTTP user agent interactions, faster page load times can increase QoE. Pages composed of multiple resources can benefit from multiplexed loading. The design and deployment considerations for HTTP/1.x pushed software towards opening multiple TCP connections to achieve multiplexing. This (ab)use of the network makes it a dominant factor in QoE while providing little information to the network about the applications needs. The result is a lowest common denominator battle between congestion and buffers.

More recently, widely deployed Internet application protocols such as HTTP/2 [20] and HTTP/3 (via QUIC) [21] have designed multistreaming as a first-class protocol property. This has allowed deployments to consolidate connections onto a single congestion controller, which offers to minimize the role that the network needs to play. The upshot of this, however, is that QoE becomes a function of multiplexing behaviour at the application implementation layer.

End-user measurements of quality of multiplexing protocols becomes an exercise in also measuring the quality of an implementation's multiplexing scheduler and congestion control algorithm. Put simply, clients are more likely to measure "how good is this webserver" rather than "how fast is my access to the Internet". Care must be taken to understand exactly what any singular measure is characterising.

To provide an example, HTTP/2 allows clients to signal dependencies and weights between streams. The intent of this is to allow clients to communicate information about their application needs to a web server. But this is just a hint that servers are allowed to ignore; they might consider other static configuration or runtime information when deciding how to respond. Therefore, an end user's QoE can depend on the user agent and server interaction at a single point in time. The variation in quality implementations of HTTP/2 prioritization, and the noticeable effects on QoE, were a factor in deciding that the protocol feature should be replaced with something simpler [22].

The Web performance community continues to develop measuring methodologies. This is often a combination of synthetic/lab testing [23] of things like bulk download time or page load time. In combination with Real User Metrics (RUM) that might be collected on a small population or collected in aggregate. As HTTP protocols have continued to evolve and use the network differently, it is important to understand how the performance of versions compare [24].

Web performance has focused predominantly on web browsing initial page loads. More recently, Google has developed Core Web Vitals as a better measure for the quality of human experience. The aforementioned collection in aggregate can be done by client-side collection of data (using for instance the W3C Navigation Timing API [25]), which is beaconed back to the operator or a partner acting on their behalf, for instance Cloudflare's Browser Insights.

Additional web page actions, like interactive experiences or watching HTTP-based video [27] are traditionally hard to measure. Many providers have defined custom metrics for applications, which are collected on the client-side and sent back to the server; CTA Common Media Client Data (CTA-5004 [28]) is one example.

Finally, there's been a history of web developers wanting access to network information (e.g. W3C Network Information API [29]) but security concerns have typically stymied those efforts.

Illusion of choice

This paper has noted that users have an illusion of choice with respect to the network. We should be under no illusion that end users are always more empowered to choose the applications that underpin their actions.

On the local user side, there has classically been freedom of choice in software. For instance, a choice of multiple user agents that all implement common standards. Interestingly, out-of-band phenomena can limit the effective choice: websites that are "best run in X" (typically indicating the developers targeting of a single browser), mobile applications that launch internal browsers (webviews) rather than allowing user selection, smart TV applications that embed a client, or platform/operating system policy that restrict the type of software that can run [30] [31] [32] [33].

On the remote application side, there's various types of choices. A service operator might choose to run their own software (picking from a large selection) on their own equipment, run in cloud computing environments, employ the services of a Content Delivery Network, and so on. However, from an end user's perspective, there's often little say in how the remote side chooses to run things.

Finally, certain types of user activity rely on use of specific applications. For instance, humans using social media or teleconferencing providers may be required to use specific application software. Although there may be provider diversity, network effects in the economical, technical or sociological axes can mean an illusion of choice [34] [35].

Conclusion

Humans use the Internet to achieve goals. Three prime parties have a stake in helping end user's achieve their Internet-based goals: client applications, the network path (or paths), and remote applications (who could also be other end users).

End users have diverse goals that can span time and space. All parties play a role in the quality of experience for these goals. Measuring the performance of each is difficult. Defining metrics that are commonly understood has some value, but understanding how the metrics of each part combine into an overall end user QoE is an exceedingly complex task. The Internet and Web communities have substantial scope to continue to improve in this area.

The role of network performance on QoE should not be underestimated. Nor should it be overestimated. A network that yields good metrics (minimises latency, loss and jitter, maximises throughput) and does so consistently provides a solid foundation for the continuously evolving needs of applications.

Applications that can access network metrics can possibly lead to indirect improvements to the network party. The opposite does not hold true — networks having more information about applications is unlikely to lead to improvements in them.

End users already have an illusion of choice. Giving them greater insight into the performance qualities of party layers underpinning their Internet activities doesn't necessarily mean they will be empowered to effect change. Collection of user data in aggregate might yield better insights for remote parties, such as application service providers, but this has to be balanced against privacy considerations.

References

[1] Mark Nottingham. 2020. The Internet is for End Users. RFC 8890. RFC Editor. https://www.rfc-editor.org/rfc/rfc8890.html

[2] UK Statutory Instruments. 2016. The Water Supply (Water Quality) Regulations 2016. UK Statutory Instruments. https://www.legislation.gov.uk/uksi/2016/614/contents

[3] Central Office of Information for Department of Transport. 1979. Pay Safe - Frisbee. The National Archives. https://www.nationalarchives.gov.uk/films/1964to1979/filmpage_safe.htm

[4] IEEE. 2014. Requirements for Harmonic Control in Electric Power Systems. IEEE-519-2014. https://standards.ieee.org/standard/519-2014.html

[5] Alan Grosskurth and Michael W. Godfrey. 2005. A reference architecture for web browsers. Software Maintenance, ICSM'05. Proceedings of the 21st IEEE International Conference.

[6] Ilya Grigorik. 2021. High-Performance Browser Networking, Chapter 14. O'Reilly. https://hpbn.co/primer-on-browser-networking/#resource-and-client-state-caching

[7] Mozilla. 2017. Enable RCWN. https://bugzilla.mozilla.org/show_bug.cgi?id=1392841

[8] Keith Moore. 2002. On the use of HTTP as a Substrate. BCP 56. RFC Editor. https://www.rfc-editor.org/rfc/rfc3205.html

[9] Mark Nottingham. 2021. Building Protocols with HTTP. draft-ietf-httpbis-bcp56bis. IETF. https://datatracker.ietf.org/doc/draft-ietf-httpbis-bcp56bis/

[10] Cloudflare. 2021. Speed Test. https://speed.cloudflare.com/

[11] Virgin Media. 2021. Speed Test. https://www.virginmedia.com/broadband/speed-test

[12] Philip Eardley, Al Morton, Marcel Bagnulo, Trevor Burbridge, Paul Aitken, Aamer Akhter. 2015. A Framework for Large-Scale Measurement of Broadband Performance (LMAP). RFC 7594. RFC Editor. https://www.rfc-editor.org/rfc/rfc7594.html

[13] Marc Linsner, Philip Eardley, Trevor Burbridge, Frode Sorensen. 2015. Large-Scale Broadband Measurement Use Cases. RFC 7536. RFC Editor. https://www.rfc-editor.org/rfc/rfc7536.html

[14] Alan Clark, Benoit Claise. 2011. Guidelines for Considering New Performance Metric Development. BCP 170. RFC Editor. https://www.rfc-editor.org/rfc/rfc6390.html

[15] Alan Clark, Qin Wu, Roland Schott, Glen Zorn. 2014. RTP Control Protocol (RTCP) Extended Report (XR) Blocks for Mean Opinion Score (MOS) Metric Reporting. RFC 7266. RFC Editor. https://www.rfc-editor.org/rfc/rfc7266.html

[16] Thomas Daede, Andrey Norkin, Ilya Brailovskiy. 2020. Video Codec Testing and Quality Measurement. draft-ietf-netvc-testing. IETF. https://datatracker.ietf.org/doc/draft-ietf-netvc-testing/

[17] ITU-T. Framework of Internet related performance measurements. 2016. Q.3960. ITU-T. https://www.itu.int/rec/T-REC-Q.3960-201607-I

[18] Vern Paxson, Guy Almes, Jamshid Mahdavi, Matt Mathis. 1998. Framework for IP Performance Metrics. RFC 2330. https://www.rfc-editor.org/rfc/rfc2330.html

[19] Barry Constantine, Gilles Forget, Ruediger Geib, Reinhard Schrage. 2011. Framework for TCP Throughput Testing. RFC 6349. https://www.rfc-editor.org/rfc/rfc6349.html

[20] Mike Belshe, Roberto Peon, Martin Thomson. 2015. Hypertext Transfer Protocol Version 2 (HTTP/2). RFC 7540. RFC Editor. https://www.rfc-editor.org/rfc/rfc7540.html

[21] Mike Bishop. 2021. Hypertext Transfer Protocol Version 3 (HTTP/3). draft-ietf-quic-http. IETF. https://datatracker.ietf.org/doc/draft-ietf-quic-http/

[22] Lucas Pardue. 2019. Adopting a new approach to HTTP prioritization. Cloudflare. https://blog.cloudflare.com/adopting-a-new-approach-to-http-prioritization/

[23] Lohith Bellad, Junho Choi. 2019. A cost-effective and extensible testbed for transport protocol development. Cloudflare. https://blog.cloudflare.com/a-cost-effective-and-extensible-testbed-for-transport-protocol-development/

[24] Sreeni Tellakula. 2020. Comparing HTTP/3 vs. HTTP/2 Performance. Cloudflare. https://blog.cloudflare.com/http-3-vs-http-2/

[25] Yoav Weiss, Noam Rosenthal. 2021. Navigation Timing Level 2 (Working Draft). W3C. https://www.w3.org/TR/navigation-timing-2/

[26] Jon Levine. 2020. Start measuring Web Vitals with Browser Insights. Cloudflare. https://blog.cloudflare.com/start-measuring-web-vitals-with-browser-insights/

[27] Roberto Pantos, William May. 2017. HTTP Live Streaming. RFC 8216. RFC Editor. IETF. https://www.rfc-editor.org/rfc/rfc8216.html

[28] CTA. 2020. Web Application Video Ecosystem - Common Media Client Data. CTA-5004. CTA. https://cdn.cta.tech/cta/media/media/resources/standards/pdfs/cta-5004-final.pdf

[29] Mounir Lamouri. 2014. The Network Information API. W3C. https://www.w3.org/TR/netinfo-api/

[30] Apple. App Store Review Guidelines. 2021. Apple. https://developer.apple.com/app-store/review/guidelines/#2.5.6

[31] Alex Russell. 2021. Progress Delayed Is Progress Denied. https://infrequently.org/2021/04/progress-delayed/

[32] Stuart Langridge. 2021. Browser choice on Apple's iOS: privacy and security aspects - A presentation for the UK Competition and Markets Authority as part of their mobile ecosystems study and investigation into Apple's app store. https://kryogenix.org/code/cma-apple/

[33] Bruce Lawson. 2021. Briefing to the UK Competition and Markets Authority on Apple's iOS browser monopoly and Progressive Web Apps. https://brucelawson.co.uk/2021/briefing-to-the-uk-competition-and-markets-authority-on-apples-ios-browser-monopoly-and-progressive-web-apps/

[34] Michael L. Katz and Carl Shapiro. 1994. Systems Competition and Network Effects. Economic Perspectives - Volume 8, Number 2, Pages 93 - 115

[35] Kate O'Flaherty. 2021. Is it time to leave WhatsApp – and is Signal the answer?. Guardian. https://www.theguardian.com/technology/2021/jan/24/is-it-time-to-leave-whatsapp-and-is-signal-the-answer