QUIC (Quick UDP Internet Connections)’s role as a transport mechanism for IP video

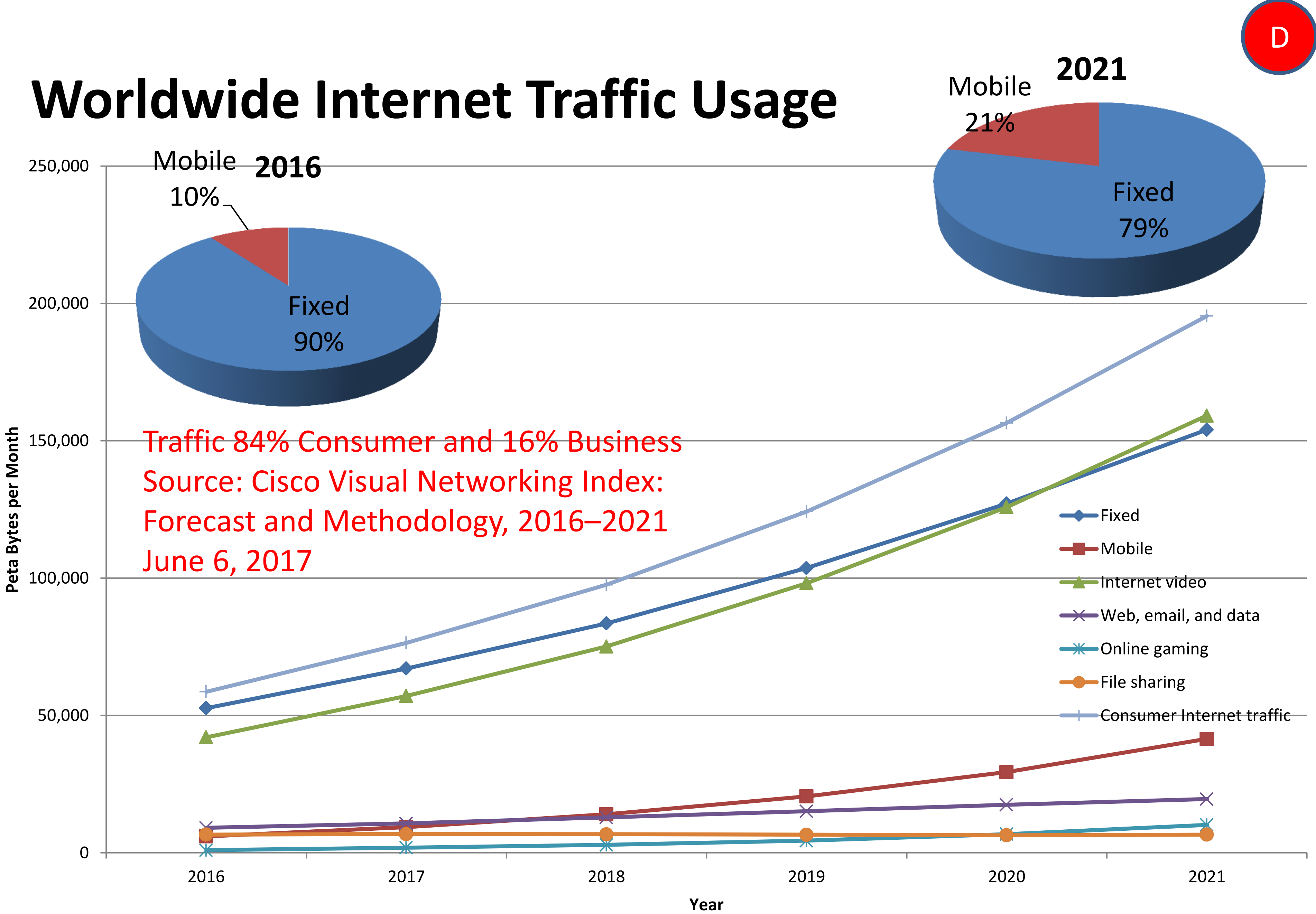
I

Contents

This is a candidate technology which will influence the future of media distribution and I'll discuss

- How networks and payloads influence the viewing experience
- Key performance criteria and strategies
- What QUIC is and why it is worthy of consideration.

Worldwide Internet Traffic Usage



How do we assess the quality of the user's experience

- Availability: the number of times the video playback starts successfully
- Abandonment Rate: percentage of viewers dropping off due to poor quality
- Start-up time: time between play button click and playback start
- Re-buffering : number of times and duration of interruptions due to re-buffering
- Bit rate: average bits per second of video playback.

D

Accelerating Bandwidth Requirements

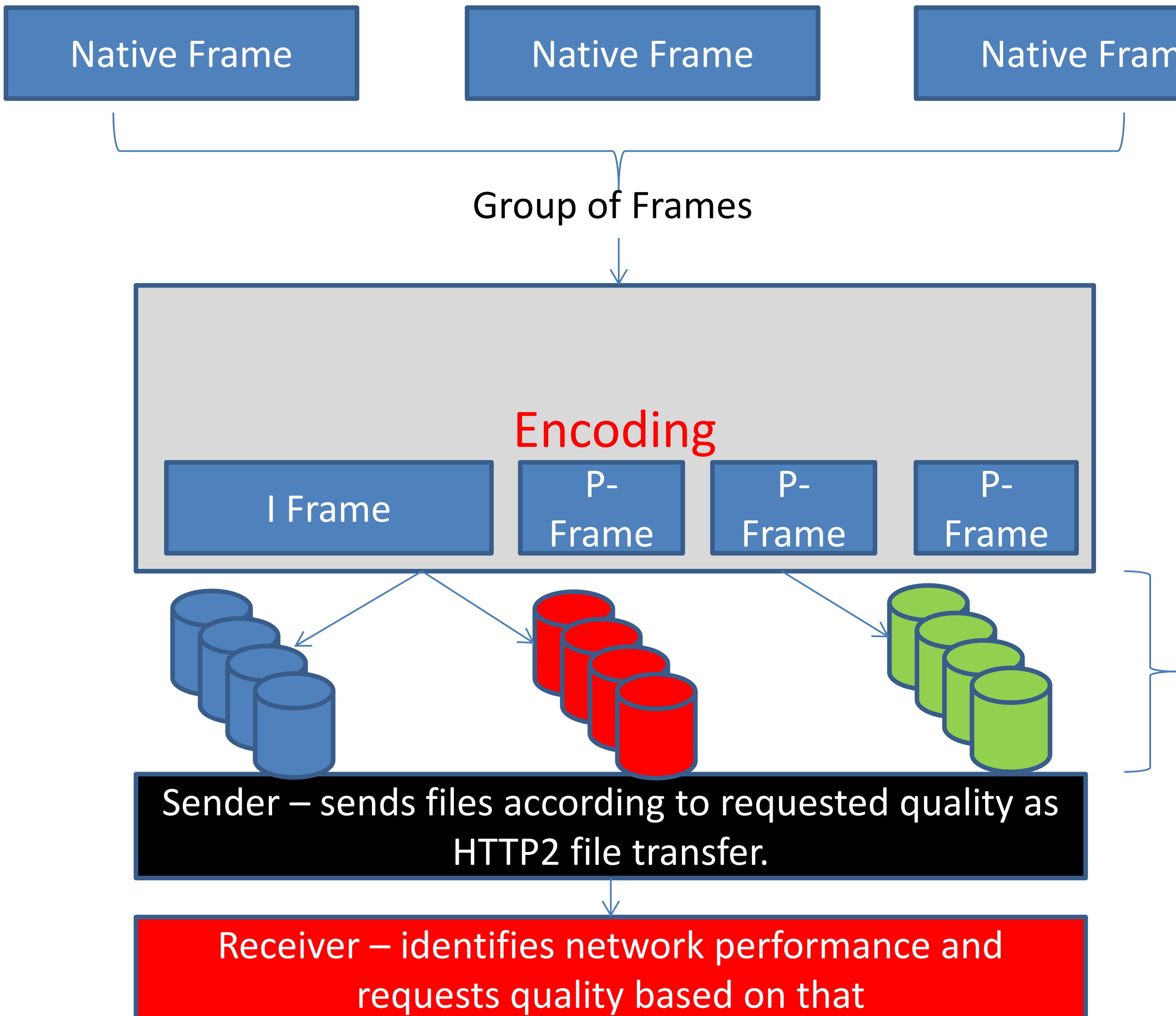
Type	Screen Resolution (Pixels)	FPS	Raw Bandwidth	With compression
SD	852 x 480	25	200M	2-3M (DVB-T)
HD	1280x470	25 or 50	622M	6-10M
4K	3840x 1080	100	12G	15-25M HEVC
8K	7860 x 4329	120	24G	100M

Data sources:

Linge,N., Murphy, L. and Darlington,W., Realising 4K: The migration to Ultra High Definition TV, Journal of the Institute of Telecommunications Professionals , 2014,8(3), pp10-13

Ilcev, D, Designing a new UHD TV standard, Journal of the Institute of Telecommunications Professionals , 2014,8(3), pp 17-20.

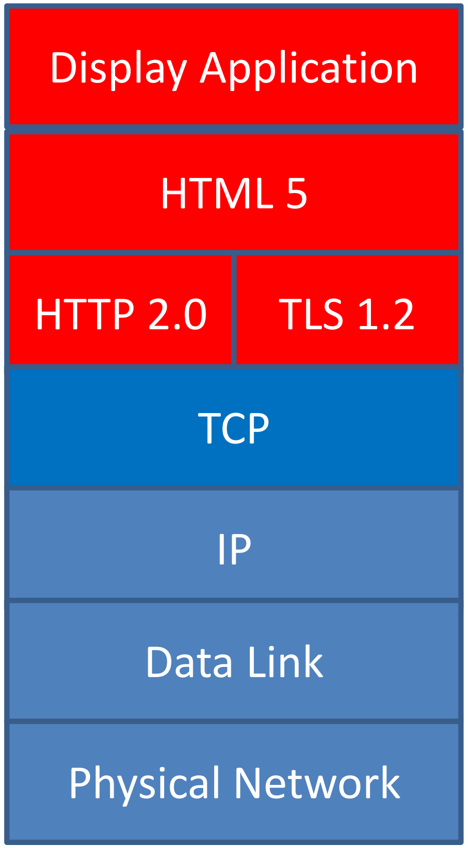
Encoding and Adaptive Bit Rate



Encoder/Decoder produces several streams to differing qualities. The goal is to send the highest quality stream commensurate with network performance

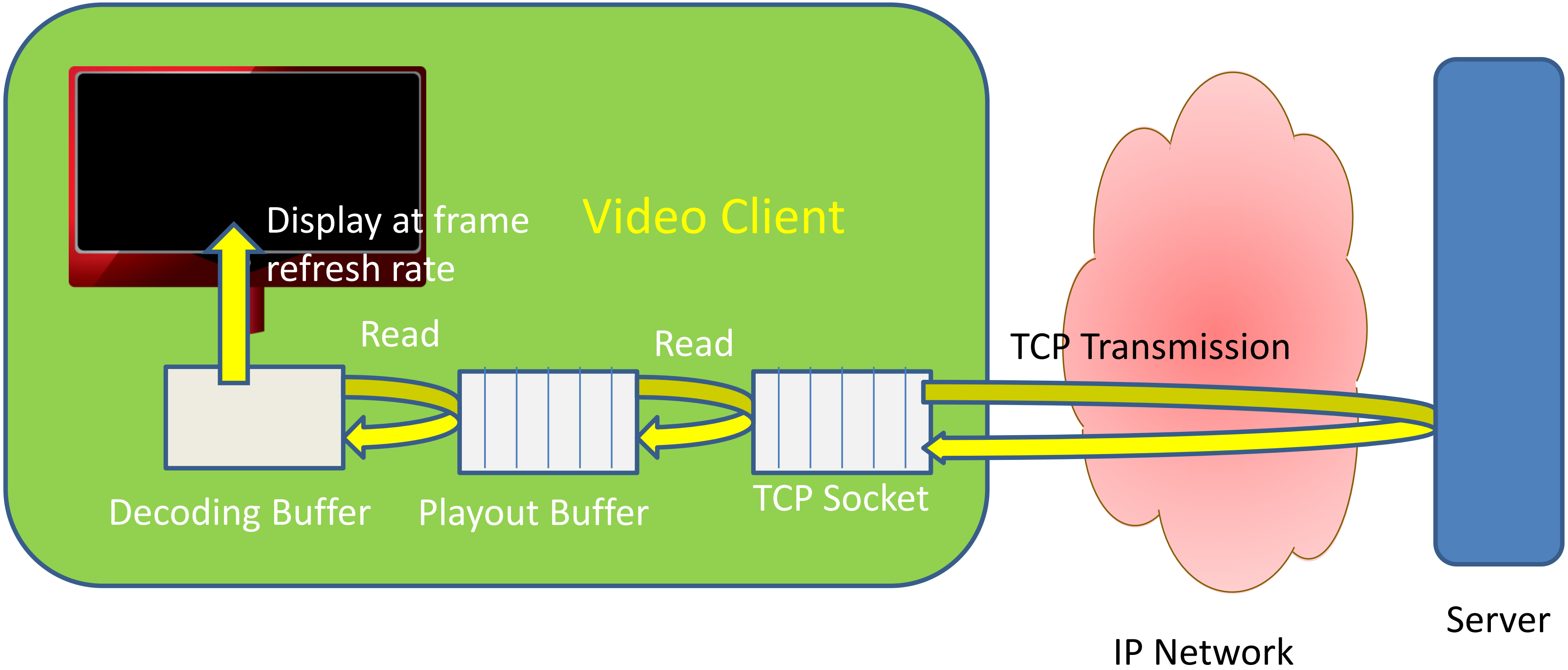
Linking this with the IP Stack

- HTML – 5
  - Development of HTML improved to support multimedia.
  - Provides interface to ABR software
- HTTP – 2 (Responsibilities taken by QUIC)
  - Allows multiplexing of messages over a single connection
  - Compression of protocol headers
  - Most client implementations use encryption (in conjunction with TLS)

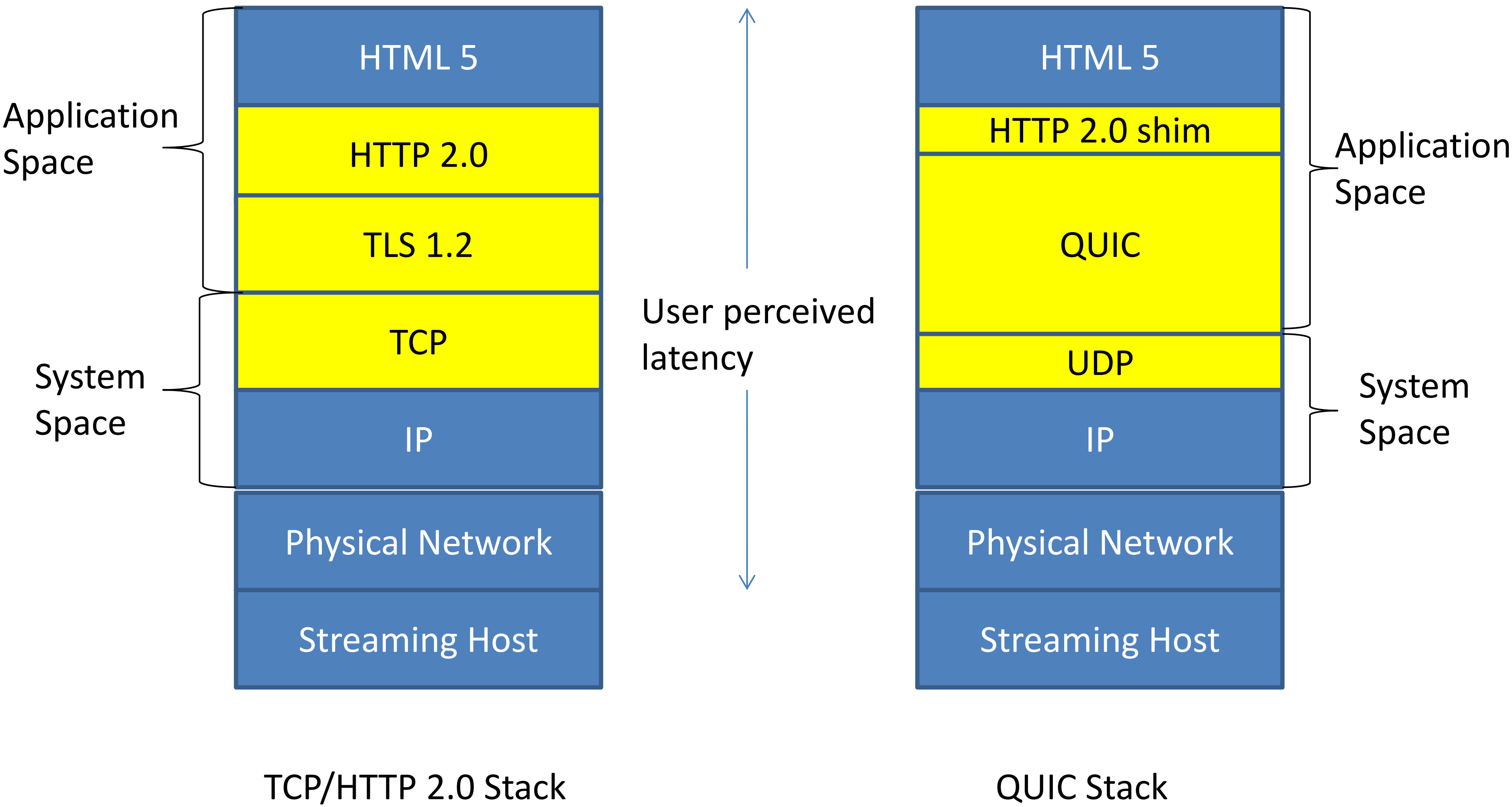


P

Buffering and Communications in IP



Video stacks



QUIC

- Invented by Google in 2012, going through IETF homologation now.
- Between 2.6 and 9.1% of Internet traffic
- Design goals
  - Deployable in user space, allowing easier modification.
  - Low latency secure connection establishment – faster setup and authentication.
  - Streams and multiplexing, removing the head of line blocking associated with TCP.
  - Better loss recovery and flexible congestion control, uses unique packet number and receiver timestamp
  - Ensure that there were no dependencies on software changes within the providers network.
  - Better management of NAT sessions and transfers between networks.

HTTP/TCP and Performance

1990

- Single static page
- 1 resource
- 1 Domain

Today

- 1200 KB web page
- 80 resources
- 30 domains

HTTP 1.1

- One TCP connection per HTTP session
- Max 6 TCP sessions per browser, more than this impairs performance.
- TCP pipelining allows multiple requests to be sent sequentially, but these have to be responded to in the same order as received. Creates Head of Line Blocking

SPDY/ HTTP-2

- Multiple HTTP sessions per connection.
- Removes the HTTP 1.1 HoL blocking.
- TCP congestion impacts all streams creating a new head of line blocking.

QUIC

- Streams run independently across protocol.
- Packet loss only impacts streams contributing to the lost packet
- Removed head of line blocking

Performance gains

- Reduced overhead in setting up transport sessions and in establishing security associations by combining handshake and caching context.
- Removal of Head of Line Blocking by only retransmitting data associated with the streams affected. Audio and video sent over two streams within a single QUIC connection.
- Improve transmission error handling:
  - Distinguish between acknowledgement of original packet and retransmission, combine unique packet number and stream offset.
  - Use of selective retransmission mechanisms.
  - Simple Forward Error Correction tried and removed in 2016
  - Better estimation of network RTT characteristics.
- Better congestion and flow control:
  - Use of HTTP 2 like, credit based, flow control mechanisms and prioritisation to reduce packet loss due to bursty traffic streams. Windowing mechanisms for connection and stream flow control.
  - Improved fast transmit mechanism.
  - Tested using the same Linux Cubic congestion control mechanism, optimised for high bandwidth with high latency networks

Performance – comparison with TCP

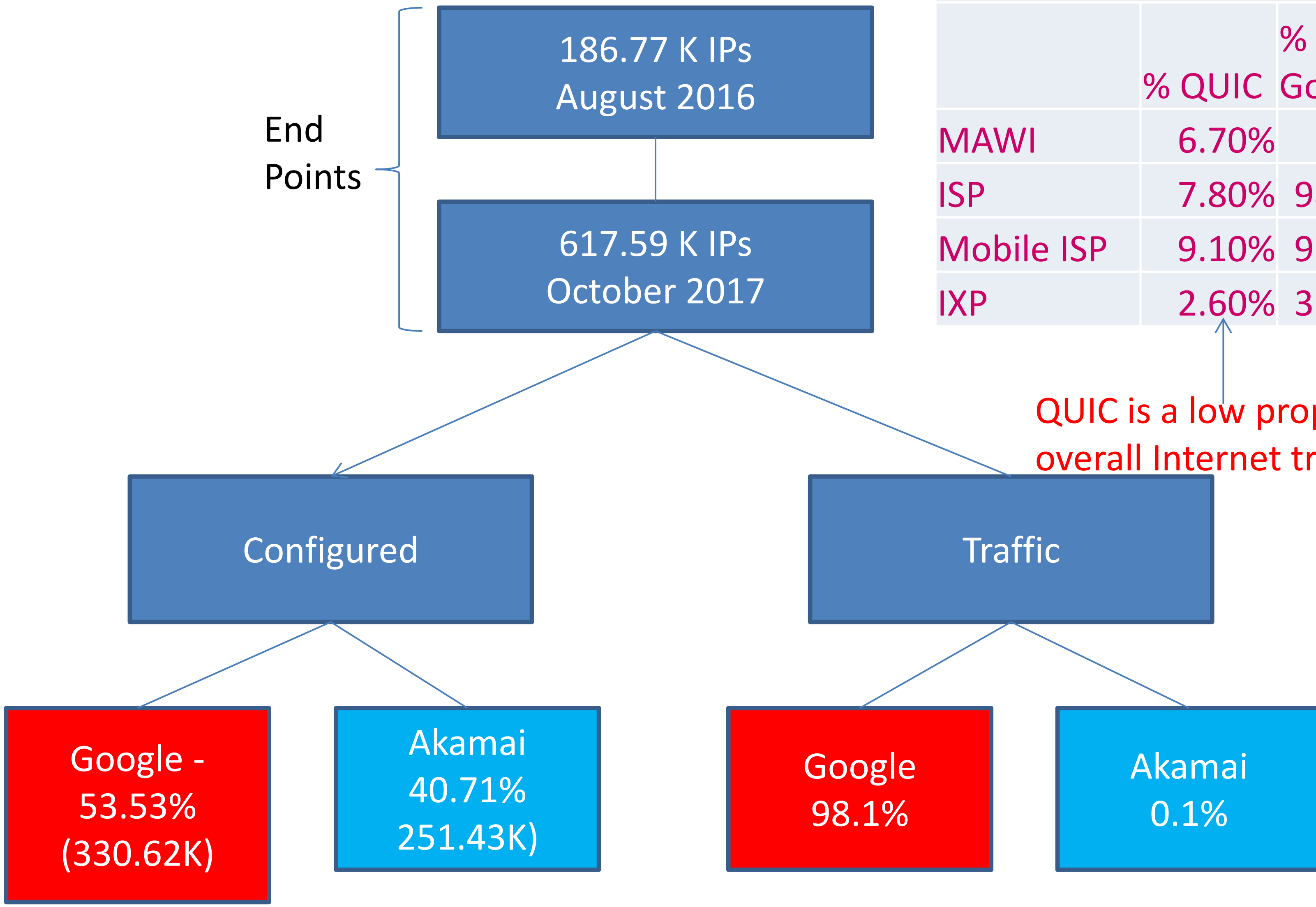
Q

- Langley et al (Google) – percentage improvement

		Mean	99 percentile
Search latency	Computer	8.0 %	16.7 %
	Mobile device	3.6 %	14.3 %
Video latency	Computer	8.0 %	10.6 %
	Mobile device	5.3 %	7.5 %
Video re-buffering	Computer	18.0 %	18.5 %
	Mobile device	15.3 %	8.7 %

  - They find server cpu utilisation doubles (reduced from 3.5) due to QUIC running in user space.
- Kakhki et al (North-Eastern/Purdue)
  - Performance depends on network and device, so improvements are variable.
  - QUIC outperforms TCP under loss in harsh environments due to improved loss recovery and lack of head of line blocking.
  - The differences are less marked for mobile devices where resources are more scarce.
  - They identify performance issues related to window size, re-ordered packets and multiplexing a large number of objects.
  - QUIC however consumes more bandwidth and starves other TCP applications
- Carlucci et al (Politecnico di Bari)
  - Similar results, includes HTTP1 in mix
  - FEC reduces QUIC throughput
  - QUIC starves TCP under congestion.
- Since these measurements were taken Google have permitted a higher maximum contention window setting, which further improves performance.

QUIC in the wild – Ruth et al



Traffic Flows			
	% QUIC	% Google	% Akamai
MAWI	6.70%		
ISP	7.80%	98.10%	0.10%
Mobile ISP	9.10%	96.90%	0.10%
IXP	2.60%	33.10%	59.90%

QUIC is a low proportion of overall Internet traffic

Security

- The level of encryption used on the Internet has risen from 15% in Jan 2012 to 85% in Jan 2016.
- Intrinsic in the protocol, headers are authenticated and most data and signalling is encrypted.
- Lychev et al:
  - Confirm soundness of QUIC security design
  - Accelerating connection establishment opens up attack vectors that could impact performance.
  - They identify and implement four threats that disrupt the handshake process and one denial of service attack. These are being assessed by Google.
- Security v monitoring impacts some Enterprises
- Can not currently be used for PCI since TLS 1.3 support is incorporated at present.
- TLS 1.3 introduction is imminent (TLS 1.3 standard ratified in March 2018)

C

Conclusions

- For streaming standardisation around IP based protocols makes sense.
- QUIC has a number of merits, but how significant are they? Depends on network quality.
- IETF drafts since 2012, slow adoption is normal as testing shakes out. User space implementation may challenge mobile devices.
- Some design decisions are to be finalised.
- Less than 10% of current web traffic
- Encryption and authentication mean no need to update middle boxes, so network upgrades are not required.
- Only implemented in Chrome and Opera, which have 60% of the browser market (https://www.w3counter.com/globalstats.php)
- Used extensively on Google's platform and Google is number one in both search and video. Akamai are a powerful ally.
- Under consideration by 3GPP for use on 5 G Packet Core.

References

- Cisco Visual Networking Index: Forecast and Methodology, 2016–2021 June 6, 2017
- Linge,N., Murphy, L. and Darlington,W., Realising 4K: The migration to Ultra High Definition TV, Journal of the Institute of Telecommunications Professionals , 2014,8(3), pp10-13
- Ilcev, D, Designing a new UHD TV standard, Journal of the Institute of Telecommunications Professionals , 2014,8(3), pp 17-20.
- Metzger, F. Rafetseder, A., Stezenbach, D. and Tutschku K. Taking a Analysis of Web-based Video Delivery, Proceedings of 50th FITCE Congress, 2011, pp 170-175
- Batalla, J.M., Krawiec, P., Beben, A., Wisniewski, P and Chydzinski, A.,Adaptive Video Streaming: Rate and Buffer on the Track of Minimum Rebuffering,, IEEE Journal On Selected Areas In Communications, Vol. 34, No. 8, August 2016, pp 2154-2167.
- Carlucci, G., De Cicco, L., and Mascolo, S. Http over udp: An experimental investigation of quic. In Proceedings of the 30th Annual ACM Symposium on Applied Computing (New York, NY, USA, 2015), SAC ’15, ACM, pp. 609–614.
- Langley, A., Riddoch, A., Wilk, A., Vicente, A., Krasic, C., Zhang, D., Yang, F. , Kouranov, F., Swett, I., Iyengar, J., Bailey, J., Dorfman, J., Roskind, J., Kulik, J., Westin, P., Tenneti, R., Shade, R., Hamilton, R., Vasiliev, V. Chang, W., and Shi, Z., The QUIC Transport Protocol: Design and Internet-Scale Deployment, SIGCOMM ’17, August 21-25, Proceedings of the conference of the ACM Special Interest Group on Data Communication, pp183-196
- Kakhki, A.M, Jero, S., Choffnes, D., Nita-Rotaru, C. and Mislove, A., Long Look at QUIC - An Approach for Rigorous Evaluation of Rapidly Evolving Transport Protocols, IMC’17, Porceings of the 2017 Internet Measurement Conference, pp 290-303
- Rüth, J. , Poese, I,. Dietzel, C. and ‘Hohlfeld, O., A First Look at QUIC in the Wild, arXiv:1801.05168v1 [cs.NI] 16 Jan 2018
- Lychev, R., Jero, S., Boldyrevaz, A. and Nita-Rotarux, C., How Secure and Quick is QUIC? Provable Security and Performance Analyses, IEEE Symposium Security and Privacy, 17-21 May 2015, DOI:10:1109/SP 2015, 21, pp 214-231