

# Evolution of QUIC and Satellite over the Last 3 Years

2nd QUIC and Satellite Open Stakeholder Meeting

Gorry Fairhurst, [Tom Jones](mailto:tom@erg.abdn.ac.uk), Ana Custura

[gorry@erg.abdn.ac.uk](mailto:gorry@erg.abdn.ac.uk) [tom@erg.abdn.ac.uk](mailto:tom@erg.abdn.ac.uk) [ana@erg.abdn.ac.uk](mailto:ana@erg.abdn.ac.uk)



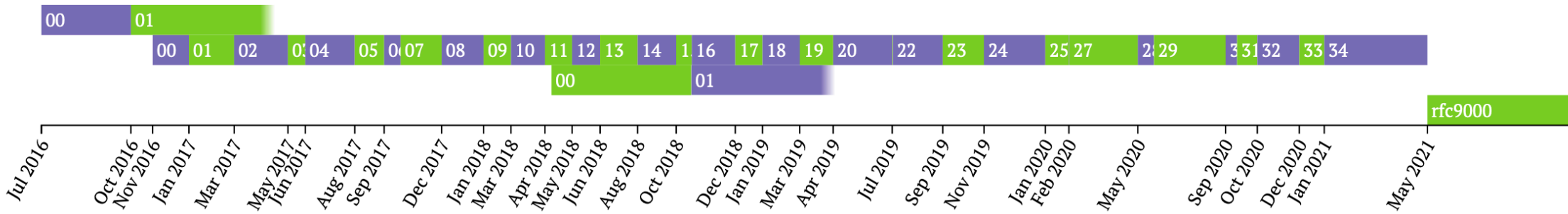
# QUIC & Satellite

In 2017 and 2018 for some QUIC spelled impending doom

TCP on GEO links was only practical when you use a PEP

QUIC's all-encrypted nature ended the benefits of these PEPs

# IETF QUIC Timeline

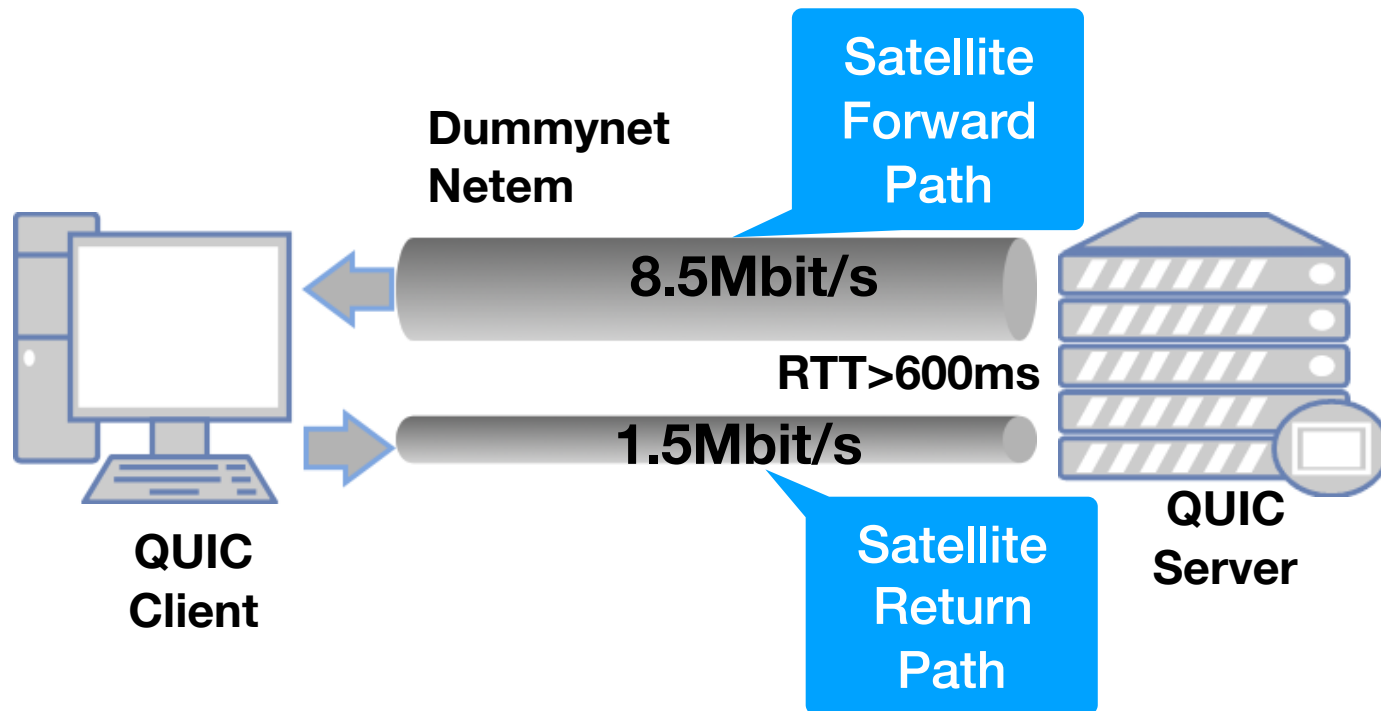


# Experimental Testbeds

Use network emulation and real links

Use QUIC as the protocol is developed, starting with quickly by fastly

Network traces and and fancy new QUIC logs



# Identified Subject Areas

Flow Control

Congestion Control

ACK Policy

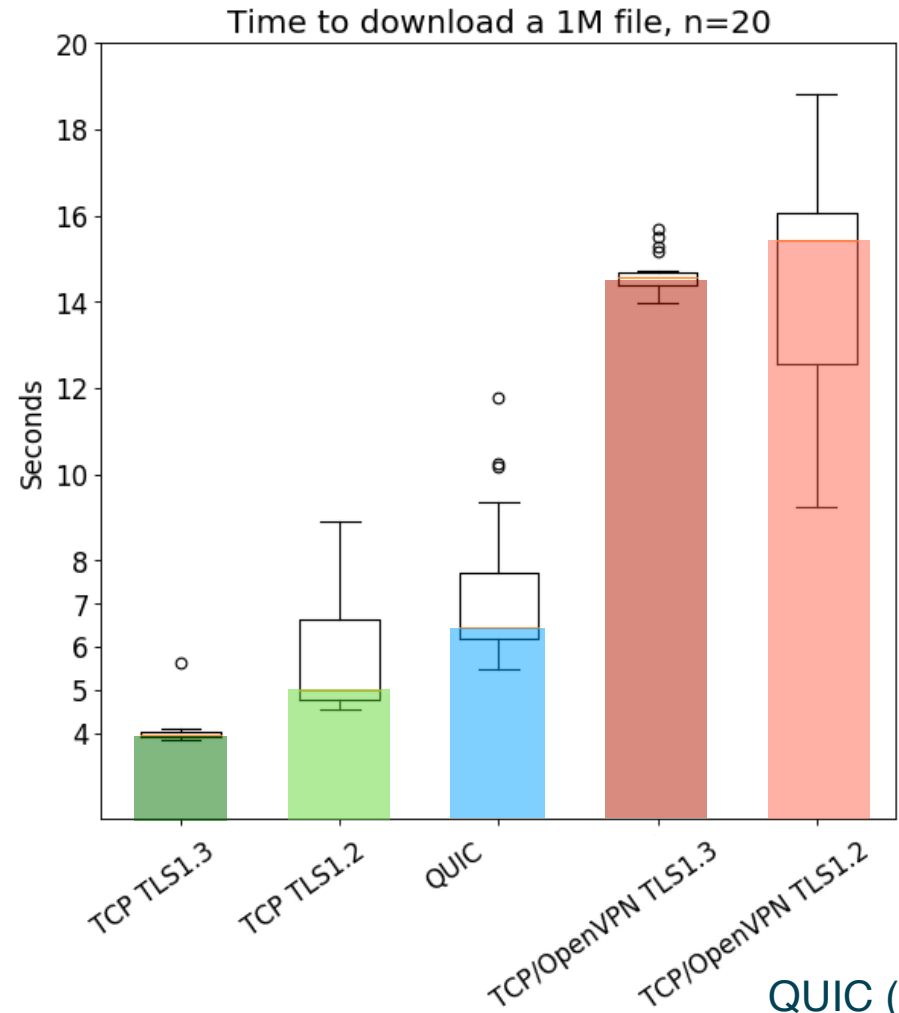
Implementation variability

# How does QUIC performance compare to TCP?

TCP with TLS 1.3

TCP with TLS 1.2

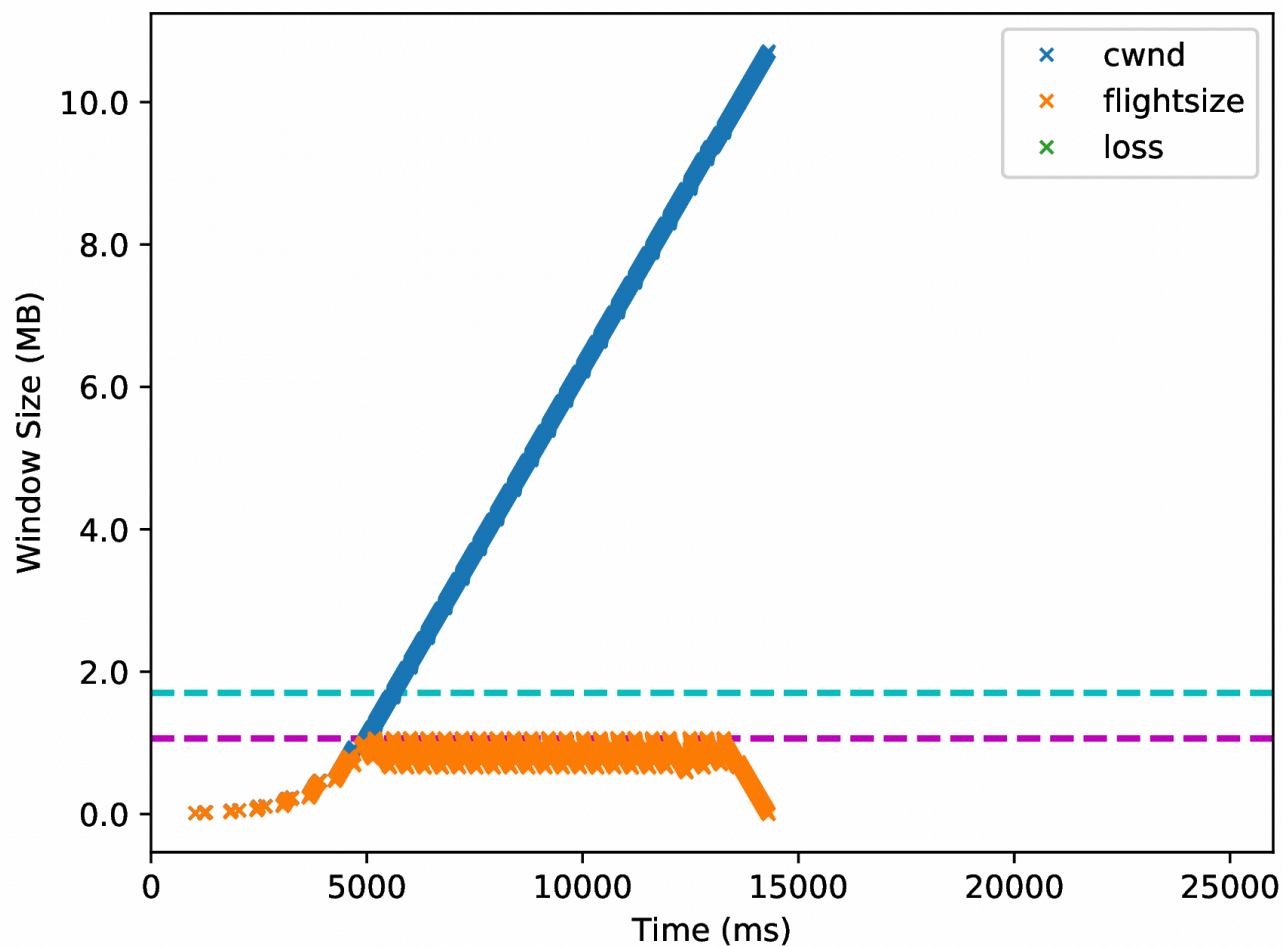
TCP inside a VPN



QUIC (quicly)

# Early Results (Summer 2019) - Flow Control Limits

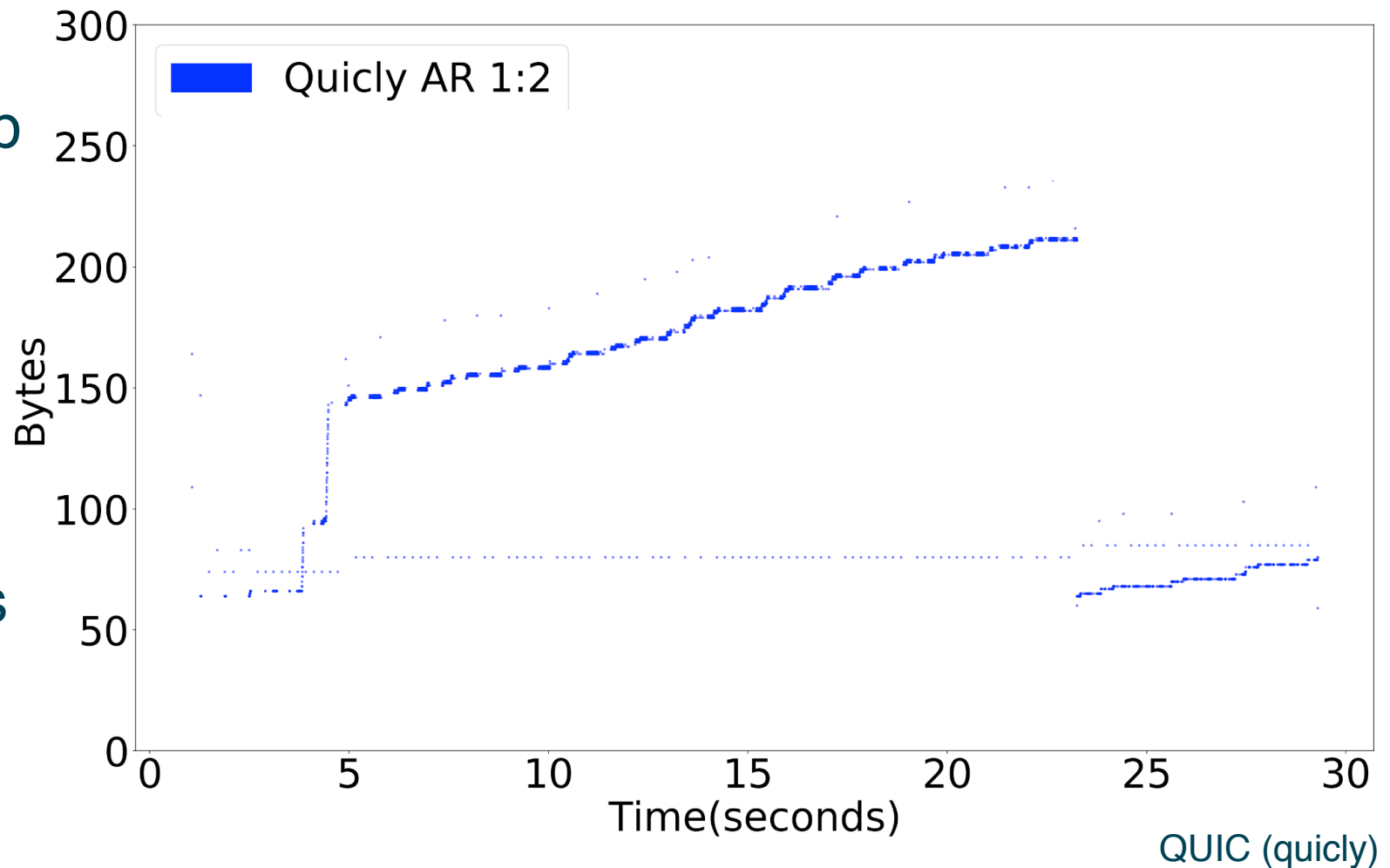
Flow Control was rate-limited



# Early Results (summer 2019) - Very Large ACKs

Observation:  
ACKs could be up to 250 bytes in length

Bug: Large size persists after loss for many RTTs





# First Recommendations

Congestion Controllers now have to deal with large BDP and high RTT

QUICs ACK Ratio is much higher than TCP, pay attention

Flow Control Needs to be designed for large BDP and high RTT

# Returning Experiment Design Advice to the Community

GEO paths are very different to common Internet paths

They are hard to develop and test for: netem and dummysnet are unintuitive

We specified characteristics of current and future GEO services in an IETF Draft

Recommendations in:

- [draft-kuhn-quic-4-sat](#)
- [draft-jones-tsvwg-transport-for-satellite](#)

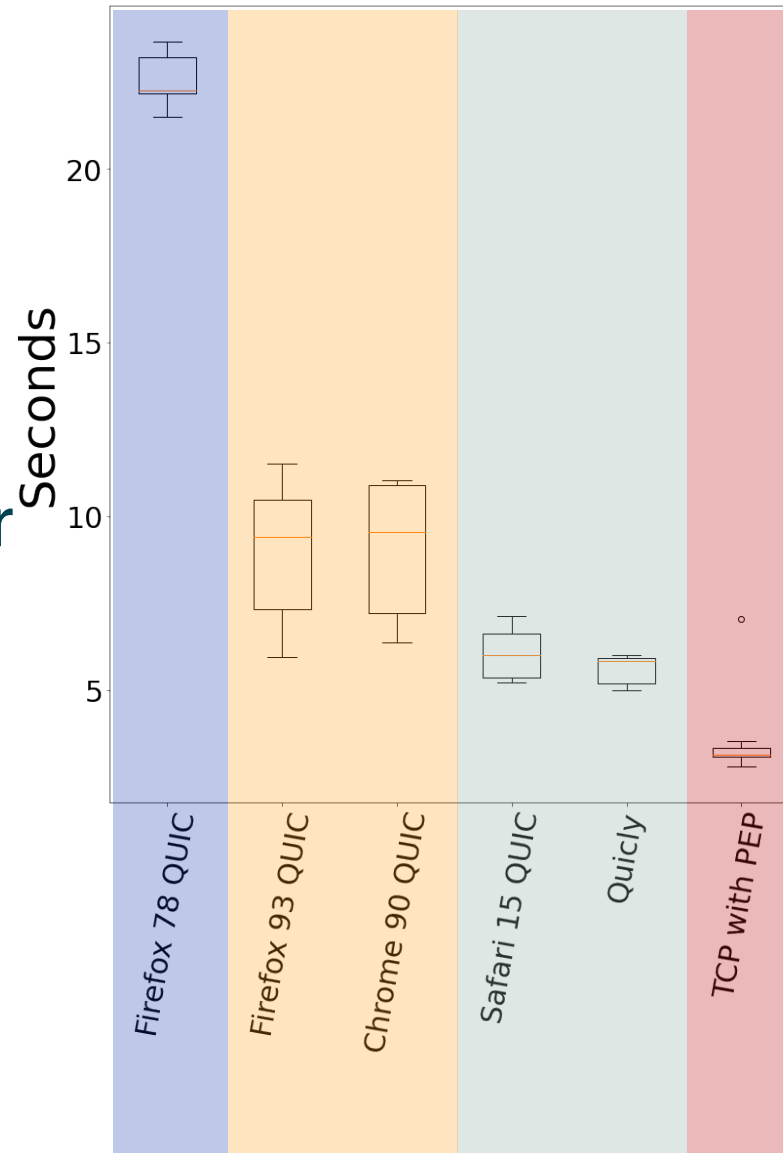
# Future

Areas that could be quicker:

CC Startup can be quicker

Loss recovery could be better

MP-QUIC possible



# Starting Up Faster

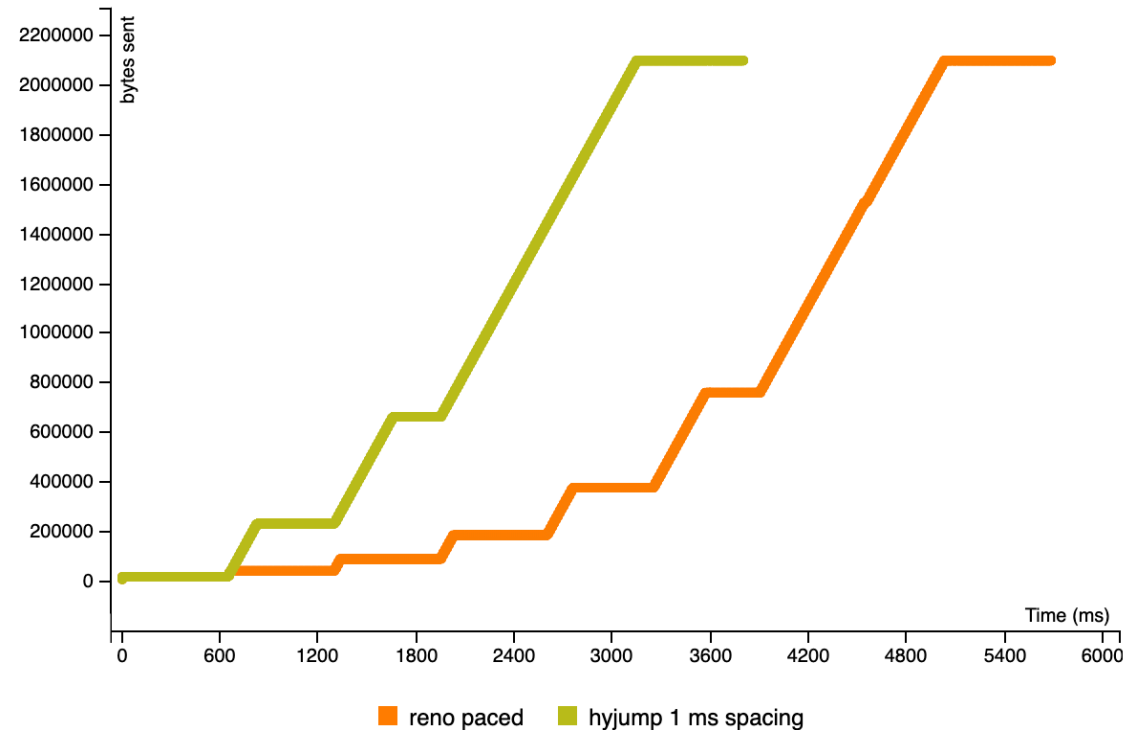
GEO send rates are 'hard won'

QUIC has great support for session resumption with 0RTT

Use the last rate to 'skip' first few RTTs of growth

see:

`draft-kuhn-quic-0rtt-bdp`



# Is QUIC getting better?

QUIC implementations are getting better

ACKs are now a QUIC work item

0RTT plot

CC

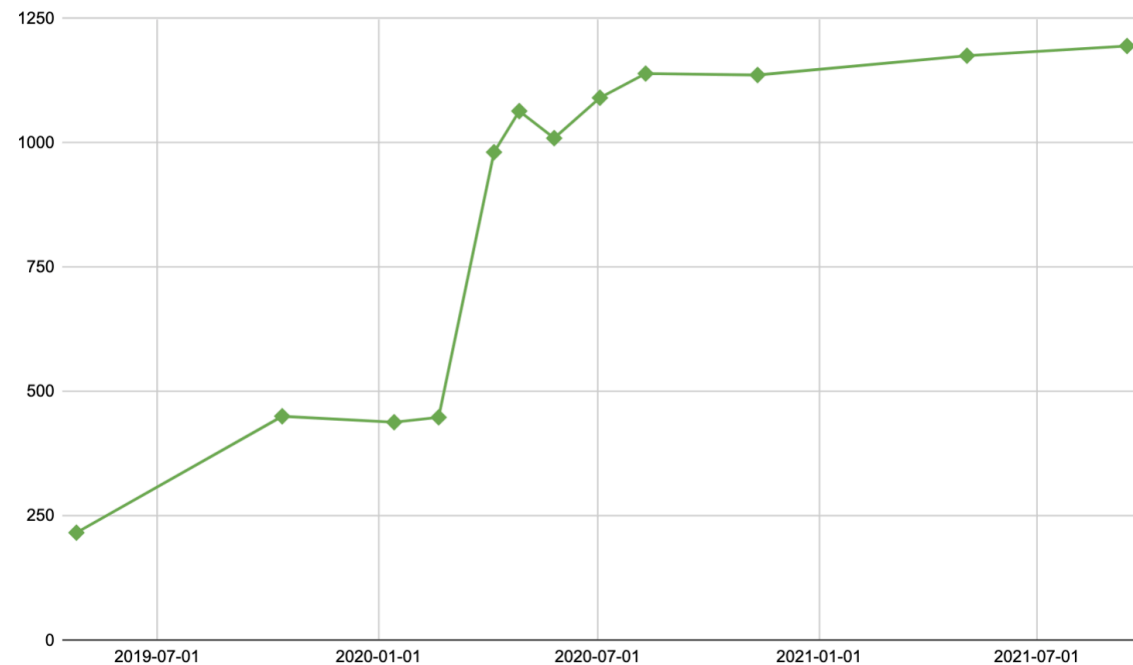
**still open:** Loss recovery (two network segments in series)

**still open:** MP-QUIC

indra



Average Data rate (Mbps)

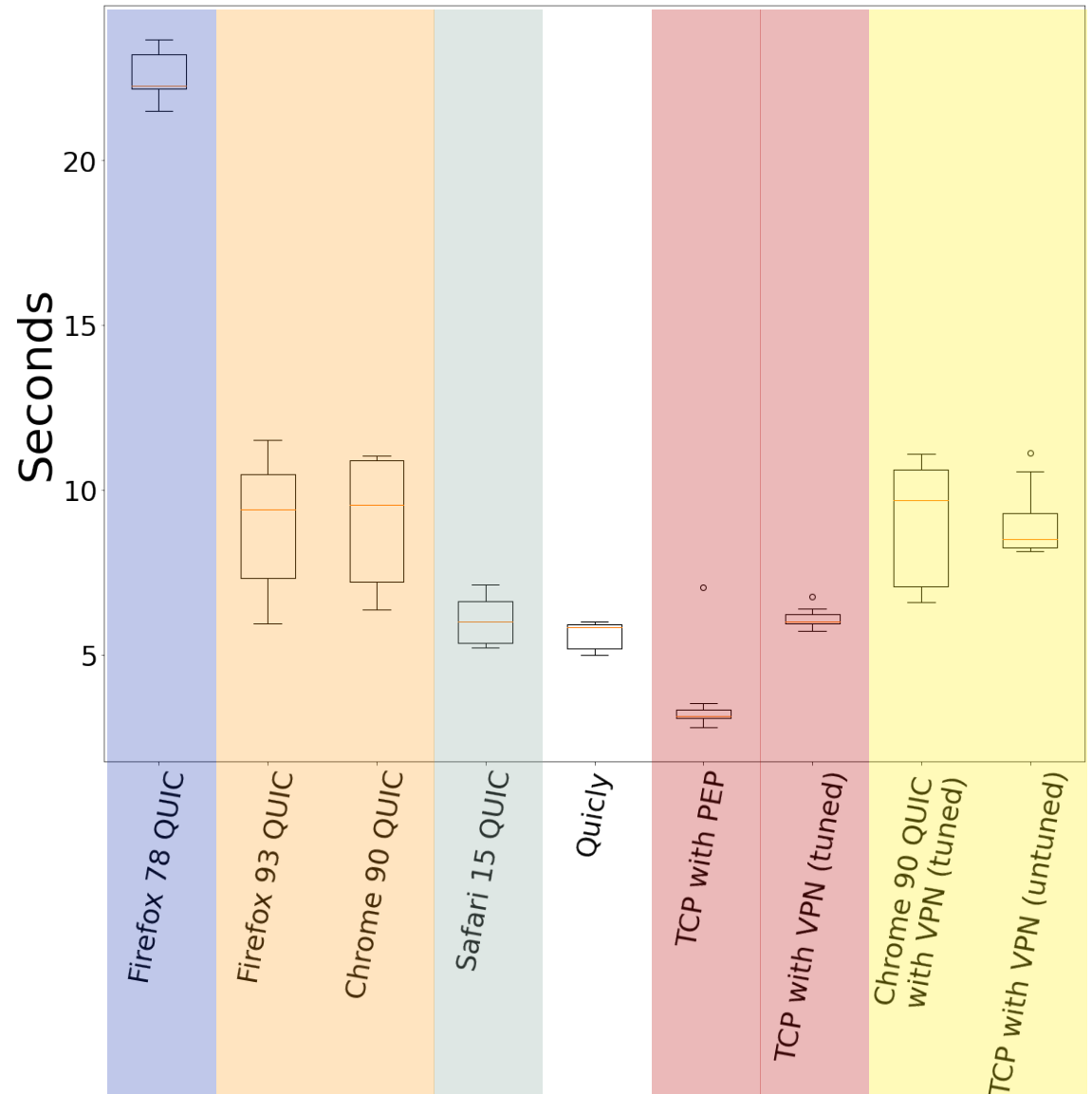


quicly: average send rate over project

# QUIC has improved in GEO networks

## QUIC continues to evolve:

- QUIC Extensions
- Uses beyond HTTP
- QUIC is better in a VPN environment than TCP!



# QUIC works in GEO Networks