Internet Architecture & Low-Latency Communications

Jari Arkko and Jeff Tantsura

May 2017

Recent Statements

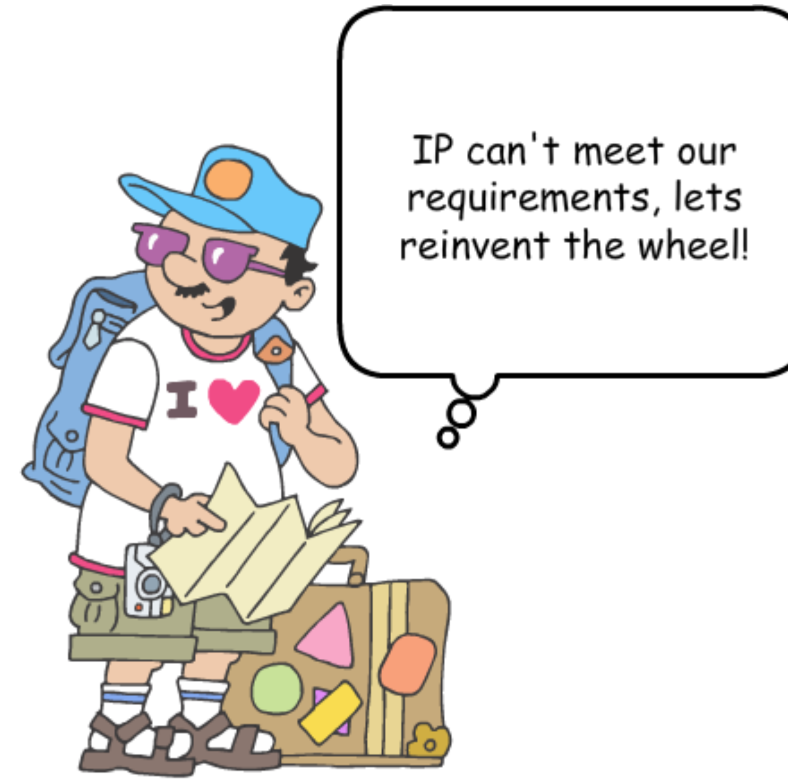
- Mission-critical 5G MTC requires low latency & high reliability and availability (Ericsson)
- Tactile Internet requires 1 ms reaction time (ITU)
- Self-driving cars require 1ms latency (Huawei)
- 5G should provide 10ms latency in the general case and 1ms in special cases, and instantaneous connection setup (NGMN)
- Should the IAB coordinate work on low latency across SDOs & investigate cross-layer interaction? (Dunbar)

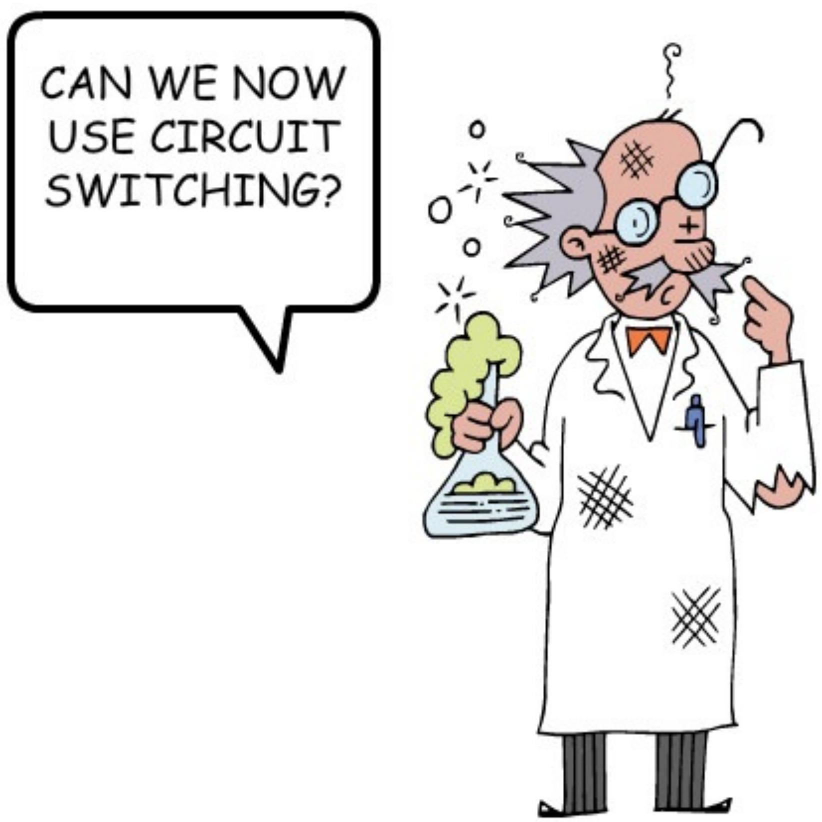
5G and low latency

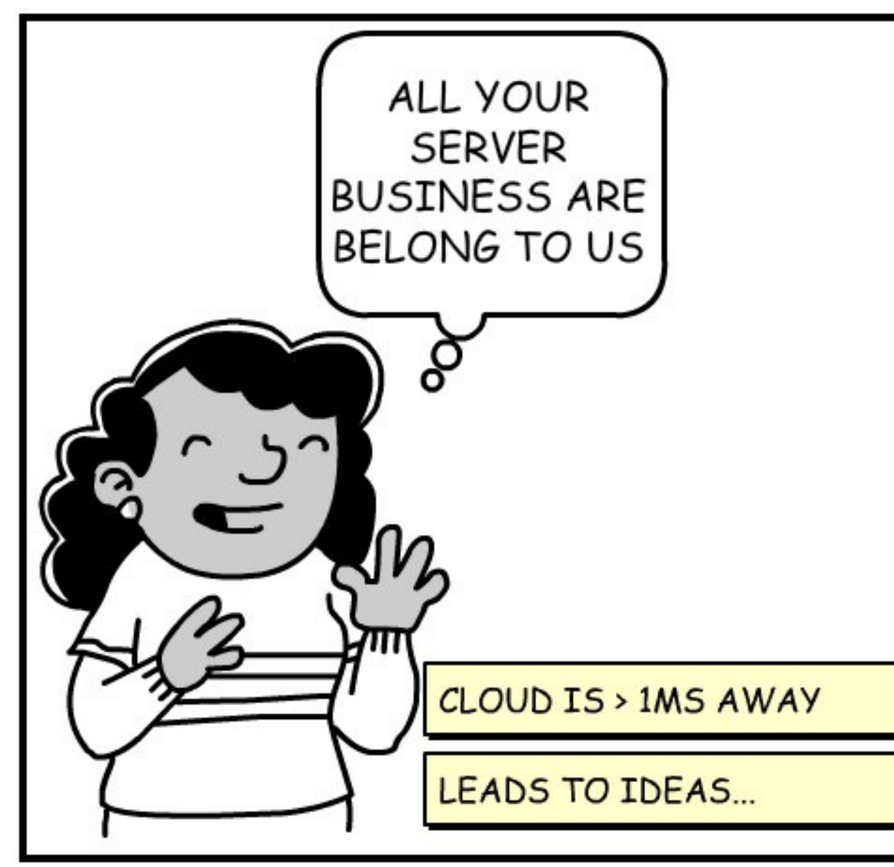
The 5G system should be able to provide 10 ms E2E latency in general and 1 ms E2E latency for the use cases which require extremely low latency. Note these latency targets assume the application layer processing time is negligible to the delay introduced by transport and switching. Use case specific E2E latency requirements are specified in Table 1.

Use case category	User Experienced Data Rate	E2E Latency	Mobility
Broadband access in dense areas	DL: 300 Mbps UL: 50 Mbps	10 ms	On demand, 0-100 km/h
Indoor ultra-high broadband access	DL: 1 Gbps, UL: 500 Mbps	10 ms	Pedestrian
Broadband access in a crowd	DL: 25 Mbps UL: 50 Mbps	10 ms	Pedestrian
50+ Mbps everywhere	DL: 50 Mbps UL: 25 Mbps	10 ms	0-120 km/h
Ultra-low cost broadband access for low ARPU areas	DL: 10 Mbps UL: 10 Mbps	50 ms	on demand: 0-50 km/h
Mobile broadband in vehicles (cars, trains)	DL: 50 Mbps UL: 25 Mbps	10 ms	On demand, up to 500 km/h
Airplanes connectivity	DL: 15 Mbps per user UL: 7.5 Mbps per user	10 ms	Up to 1000 km/h
Massive low-cost/long-range/low-power MTC	Low (typically 1-100 kbps)	Seconds to hours	on demand: 0-500 km/h
Broadband MTC	See the requirements for the Broadband access in dense areas and 50+Mbps everywhere categories		
Ultra-low latency	DL: 50 Mbps UL: 25 Mbps	<1 ms	Pedestrian
Resilience and traffic surge	DL: 0.1-1 Mbps UL: 0.1-1 Mbps	Regular communication: not critical	0-120 km/h
Ultra-high reliability & Ultra-low latency	DL: From 50 kbps to 10 Mbps; UL: From a few bps to 10 Mbps	1 ms	on demand: 0-500 km/h
Ultra-high availability & reliability	DL: 10 Mbps UL: 10 Mbps	10 ms	On demand, 0-500 km/h
Broadcast like services	DL: Up to 200 Mbps UL: Modest (e.g. 500 kbps)	<100 ms	on demand: 0-500 km/h

But There Area Also Wilder Claims







This changes in no way the dynamics and economics of Internet evolution

…

Many of those changes are unlikely, although some may lead to new business (e.g., edge computing)

But There **Is** Evidence that the World Cares about Low-Latency

- Data centers distributed around the globe
- Including content served from operator premises
- Advanced optimisation techniques for connecting to data centers (DNS etc)
- Industry working HTTP2, QUIC, TLS.1 (0-RTT), L4S, DETNET, 802.1 TSN, 5G radios, …
- SDN and SFC replacing long chains of processing functions
- Industry working on ServiceWorker, AMP, …

Lets Recap To Be Clear

- Latency in L2 is being improved
- Latency in routing/forwarding is being improved
- Latency in transport is being improved
- Latency in security is being improved
- Latency in application protocols is being improved
- Network deployments are changing to take into account latency

And it is all part of our regular program anyway

So?

But, All is NOT Done and Not Only Bad Ideas

- Obviously much of this is work in progress
- Some of it may also require coordination
  - Uncoordinated changes at different layers are very likely to create racing conditions and make e2e latency worse
- But, more importantly, the Internet is changing and this may cause strain for the architecture

Architectural Pressures 1/3

- Placing of services in different locations in the network
    - From global datacenters to more regional ones (already done in many cases anyway)
    - Possible further pushes with edge computing?
    - Additional co-operative solutions between network providers, CDNs, and content providers?
- Impacts on evolution of architectures that employ tunnelling
    - Dynamically chosen tunnel server locations, local breakout, completely new mobility architectures
    - Security implications of local breakouts – decap/encap in the middle
    - Unwillingness to deploy security measures necessary due to added latency
- There are and will be demands on cross-layer optimisation, is that a good thing for the architecture and its flexibility?
    - Data normalization (data modeling) is of high importance as needed to facilitate cross-layer conversation

Architectural Pressures 2/3

- Choice between completely local designs (e.g., cars braking and informing nearby others cars) and designs with actual networks or connectivity to the Internet
- Designing applications entirely in their own silo vs. applications that also talk to peers across the Internet
  - Everything happens in a low-latency special “slice”?
  - But we have automation systems, factories, airplane networks that do need low-latency communications between components, but also need to talk to software update servers, manufacturer maintenance server, …
- Tension between application/edge and network control of forwarding decisions (e.g., MPTCP vs. traditional routing)
- 1-bit of information to help network make forwarding decisions (Marnew, Accord, …)

Architectural Pressures 3/3

- Deployment story for new QoS or low-latency tech
  - On Qos, Dave Clark's article gives a very pessimistic view of QoS deployments… (https://www.caida.org/publications/papers/2015/adding_enhanced_services_internet/adding_enhanced_services_internet.pdf)
  - Basically, tech is not enough, also have to get the ecosystem to agree on how costs/rewards are split
  - Do low-latency deployments have some of the similar aspects, or not?
- Inter-organisatorial matters, e.g., to what extent different standards organisations need to talk about low latency effects and ongoing work