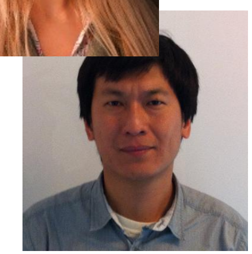
# L4S: Ultra-Low Queuing Delay for All


**Bob Briscoe**, Olga Bondarenko

[ simula . research laboratory ]

Koen De Schepper, Inton Tsang

NOKIA Bell Labs

Jun 2017

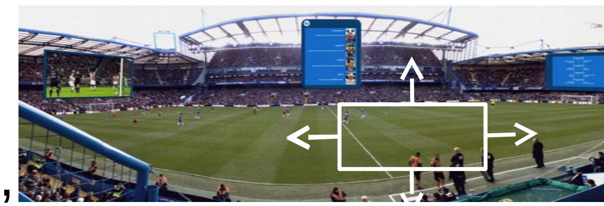
RITE
Reducing Internet Transport Latency

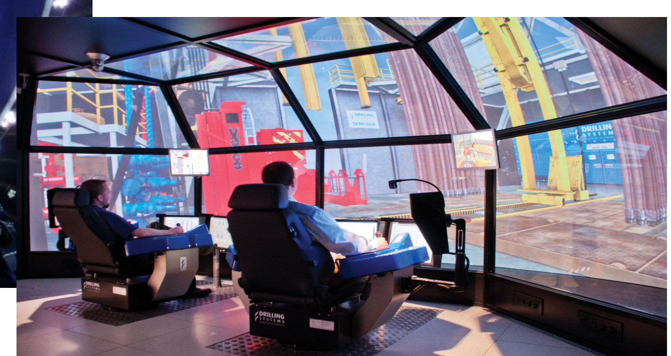
The authors' contributions are part-funded by the European Community under its Seventh Framework Programme through the Reducing Internet Transport Latency (RITE) project (ICT-317700). The views expressed here are solely those of the authors.

SEVENTH FRAMEWORK PROGRAMME

# application profile is evolving

- increasingly nearly *all* apps require low delay (and often high bit rate too)
  - interactive web, web services
  - voice,
  - conversational video, interactive video, interactive remote presence
  - instant messaging
  - online gaming
  - virtual reality, augmented reality
  - remote desktop, cloud-based apps
  - video assisted remote control of machinery & industrial processes



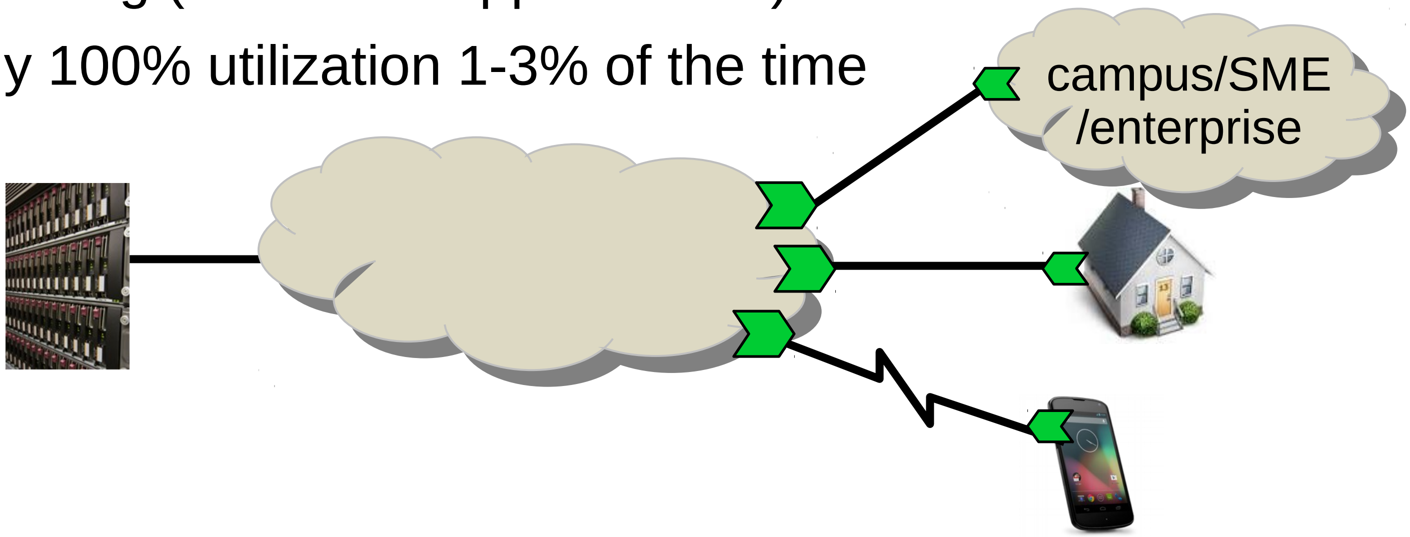



# Main contributions to delay

- Delay: multifaceted problem [Briscoe14]
  1) Caches have cut base (speed-of-light) delay, where they can
  2) Remaining major component of delay: queuing
     - intermittent – solely under load
     - at best, doubles the base delay

# Problem characterization

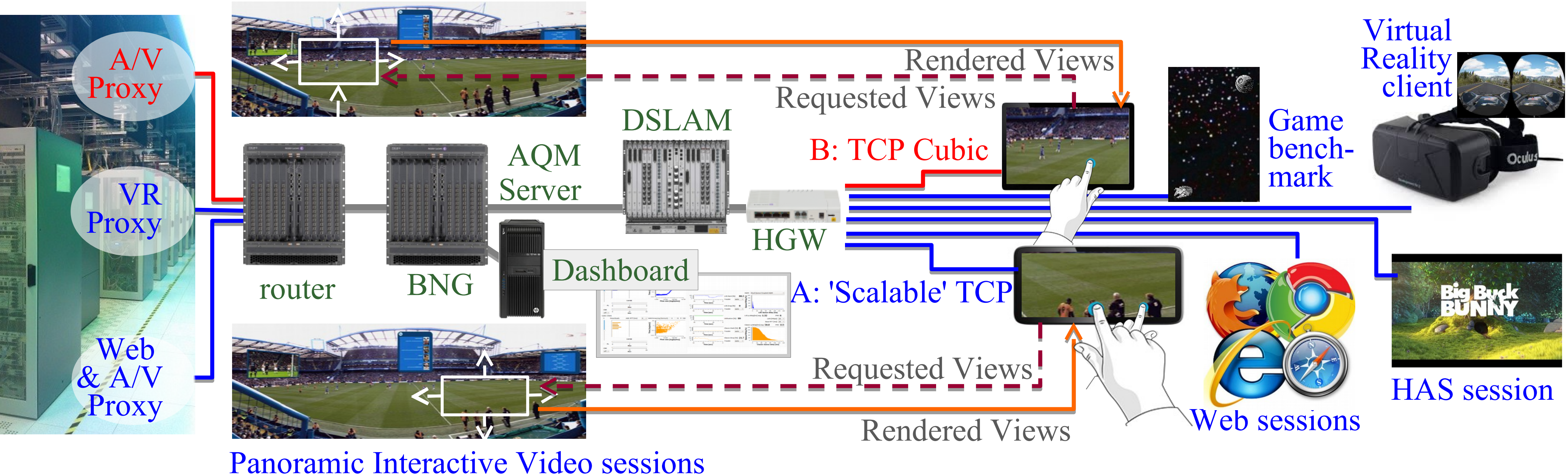
- bottlenecks typically at access edge – per 'customer'
- low statistical multiplexing (1 or a few apps at once)
- during peak, typically 100% utilization 1-3% of the time




- The new norm: all of a customer's apps at any one time need low delay
- Diffserv (Differentiated Services)?
  - better delay at the expense of the customer's other apps (queue jumping)
  - increasingly inapplicable

# Solution: L4S
## Low Latency, Low Loss, Scalable throughput

- Demo schematic
  - multiple demanding applications over the same broadband line (40Mb/s downstream)
  - plus ~4 large file downloads
  - 7 ms base (speed-of-light) round trip time from Data Centre to the Home and back
- mean per-packet L4S queuing delay ~500$\mu$s (½ ms)

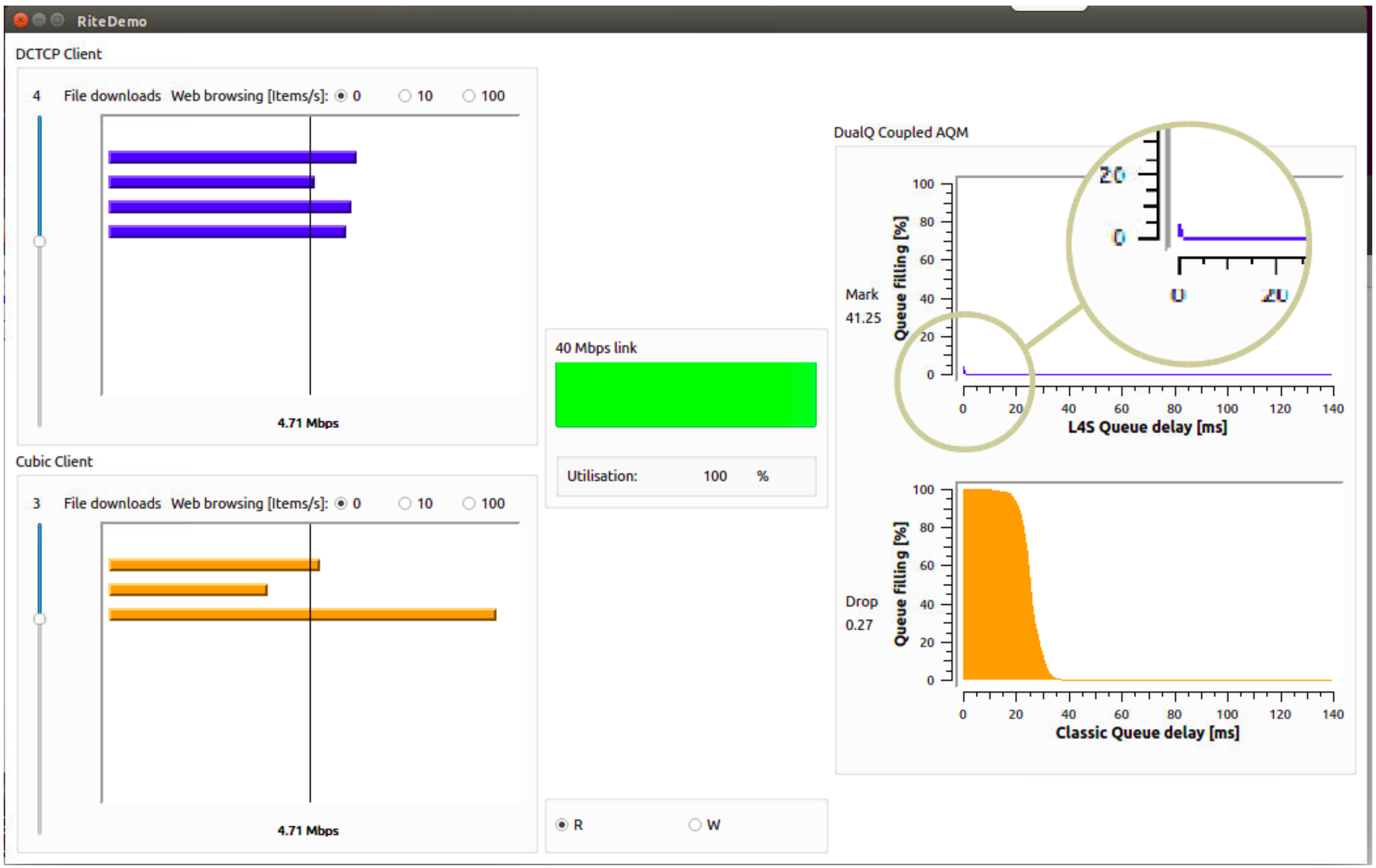



- See video of Panoramic Interactive Video + TCP downloads: https://riteproject.eu/dctth/#1511dispatchwg

# Simplified dashboard

L4S

Classic




Flow rates (window)

Link Utilization

Distribution of queuing delay

# Business Implications of “Low Delay for All”

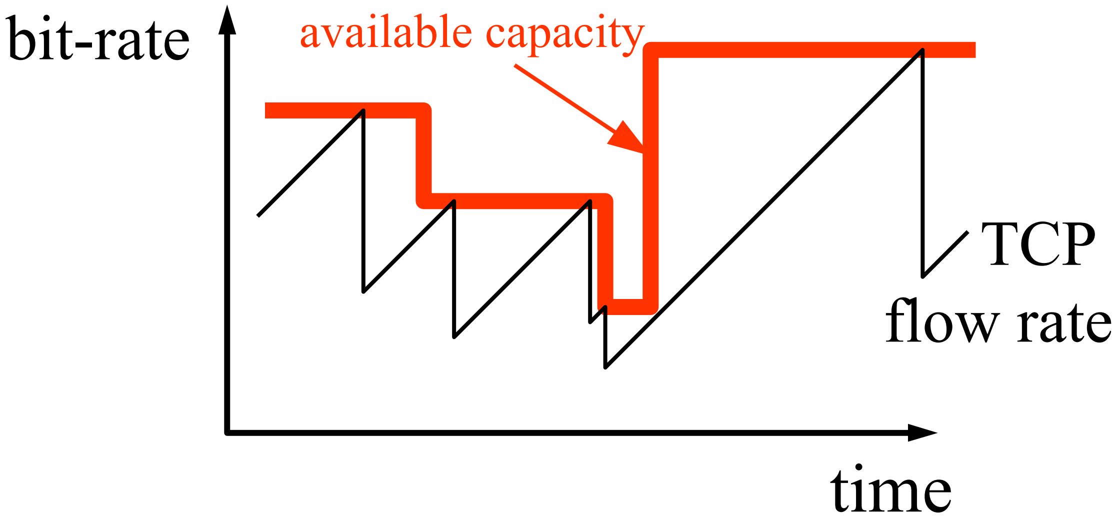
- can sell L4S per customer rather than per app
  - e.g. small businesses, premium users, enterprises (where migrating from VPN to Internet)
- eventually (or from the start) to all users

# The Solution

Remove the Root-Cause of Queuing Delay

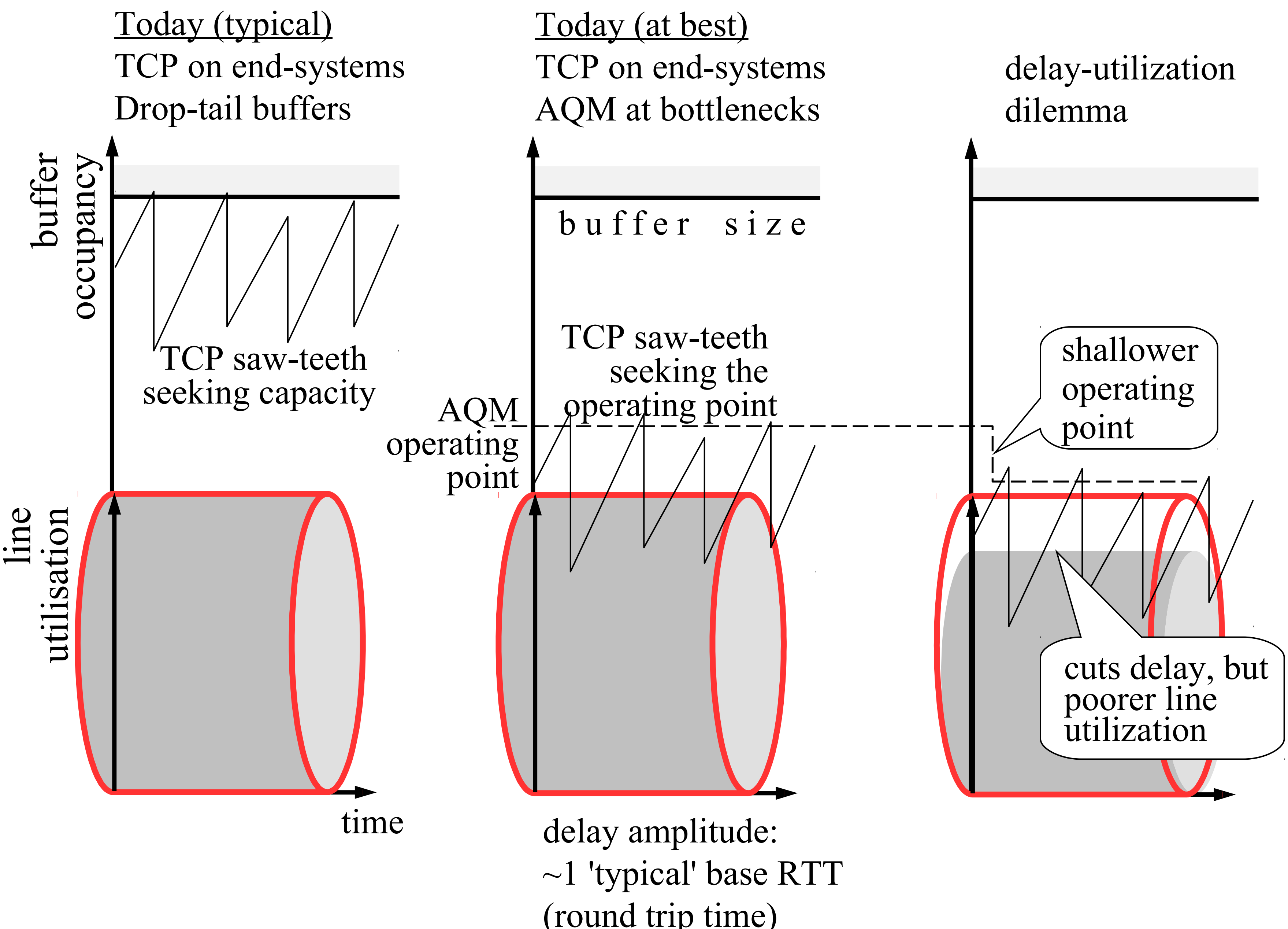
# The Cause of Queuing Delay

- TCP's capacity-seeking behaviour: 'saw-toothing'

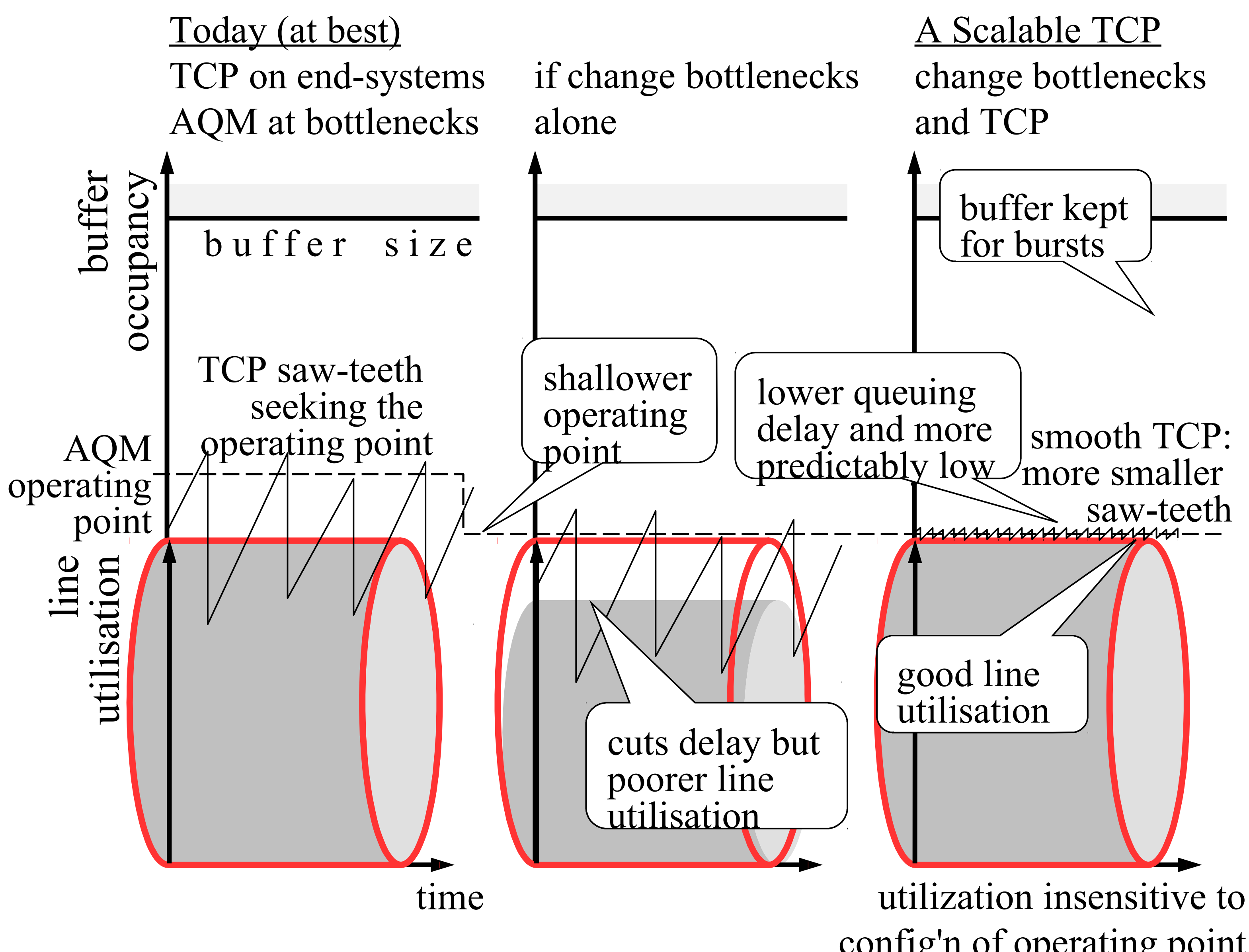



- Note on terminology:
  - Capacity seeking is called 'congestion control'
  - the outcome is called 'congestion' even when just one flow

# Active Queue Management (AQM) dilemma: delay vs. utilization



# Resolving the dilemma:
## Finer saw-teeth of a 'Scalable' TCP



# Finer Sawteeth → more frequent sawteeth

- Packet drop: no longer feasible as a congestion signal – too much
  - use Explicit Congestion Notification (ECN)
  - standard part of the Internet Protocol (v4 & v6) since 2001

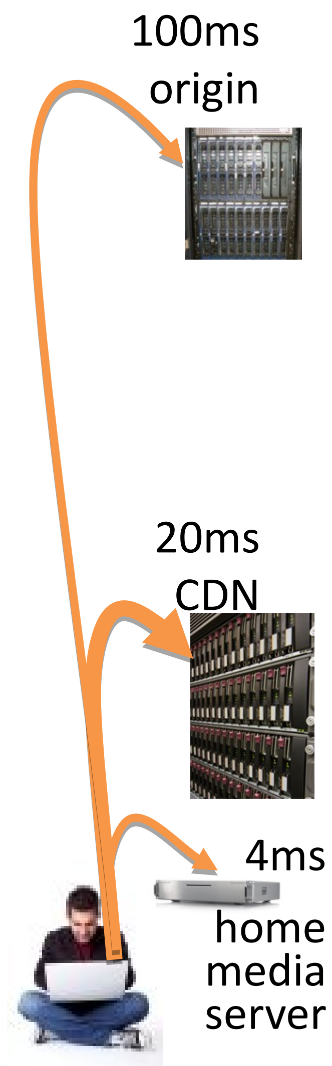
ECN: an opportunity to remove much more delay...

## Drop

- state-of-the-art AQMs defer dropping for ~100ms (*worst-case* RTT)
  - in case the burst clears itself
  - no response for 5 CDN RTTs, or 25 media server RTTs

## ECN

- AQM can signal ECN immediately – no risk of impairment
- the sender can smooth out ECN signals (over its *own* RTT)
  - can react without smoothing if appropriate




RTT: Round Trip Time

# Finer Sawteeth → more frequent sawteeth

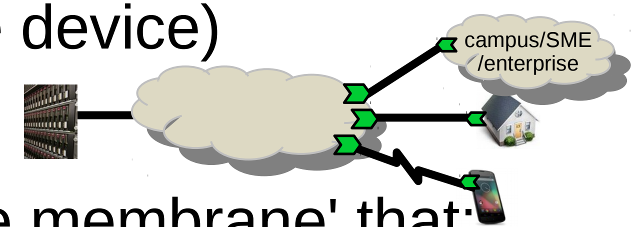
- Only feasible:

1) with modified TCP/ECN feedback
   - standard TCP source only responds to one signal /RTT
   - so when ECN was added to TCP, only one feedback /RTT
   - IETF now standardising accurate TCP ECN feedback AccECN [RFC7560, Briscoe17]

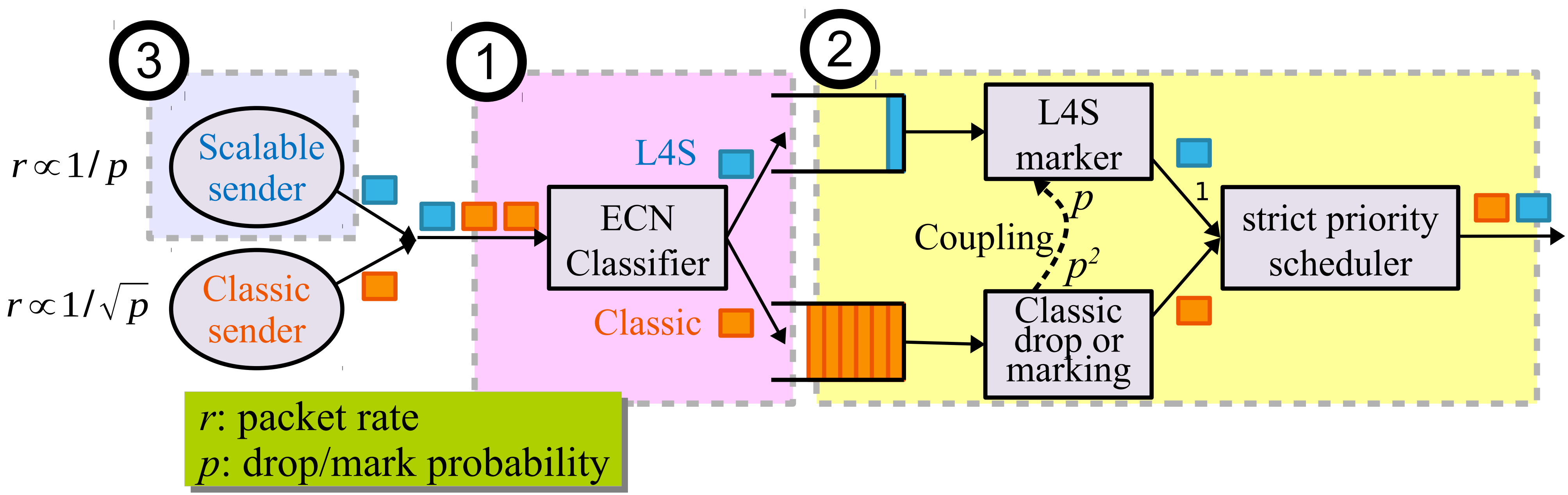
2) if coexistence with existing traffic is solved
   - a Scalable flow with fine saw-teeth
     looks like high congestion to a 'Classic' TCP flow
   - so Classic TCP starves itself relative to a Scalable TCP

# Coexistence: Solution

- per customer site (home, office, campus or mobile device)




- DualQ Coupled AQM 1 & 2: a 'semi-permeable membrane' that:
  - isolates latency (separate queues for L4S & Classic)
  - but pools bandwidth (shared by apps/transport, not by network)





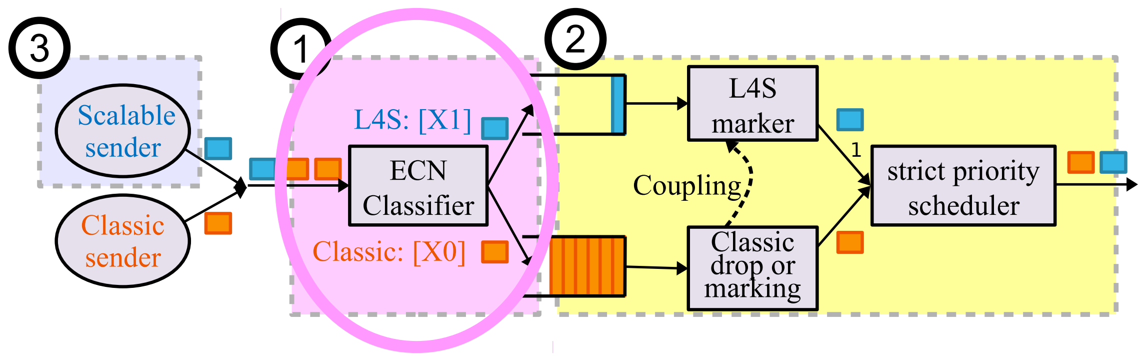

Details: [l4s-arch], [dualq-aqm]

# Other solutions - in context

- Priority classes (Differentiated Services)?
  - favouring certain traffic requires policing and management
  - non-solution when *all* of a customer's apps at one time need low latency
- Active Queue Management
  - a solution 'for all' – promising direction
  - but TCP is (literally) the elephant in the room – min queue 5-20ms
- Per-flow queuing?
  - isolates each flow from the delay of others, but overkill...
    1. removes control from variable bit rate apps (network schedules the queues)
    2. individual app flows not always visible to network (e.g. IPsec VPN)
    3. computationally expensive
    4. anyway, doesn't protect a flow from the delay it inflicts on *itself*
- BBR (Google research)
  - Attempt to reduce queuing delay without changing network
  - Queuing delay intermittent - similar to AQM
  - Problems interacting with AQM: toggles between starving others or itself
- 'Classic' (standard) ECN
  - A congestion 'mark' is equivalent to a packet drop
  - Removes round trip delay delay to repair congestion loss, but not queuing delay

# Premium Service vs. Default?

| Codepoint | ECN bits |
|---|---|
| Not-ECT | 00 |
| ECT(0) | 10 |
| ECT(1) | 01 |
| CE | 11 |




- Classifier on 2-bit ECN field in IP header (v4 or v6)
  - if ECT(1) or CE, forward to L4S
  - adopted for standardisation by IETF

**AND**
optionally

- Classifier on any other field
  - source IP address
  - dest. IP address
  - VLAN ID, ...

- ECN field works end-to-end
  - network could solely enable L4S for certain addresses
  - later, could enable for all addresses
- in all cases, no packet inspection deeper than IP
  - compatibility with all privacy technology

Details: [l4s-id]

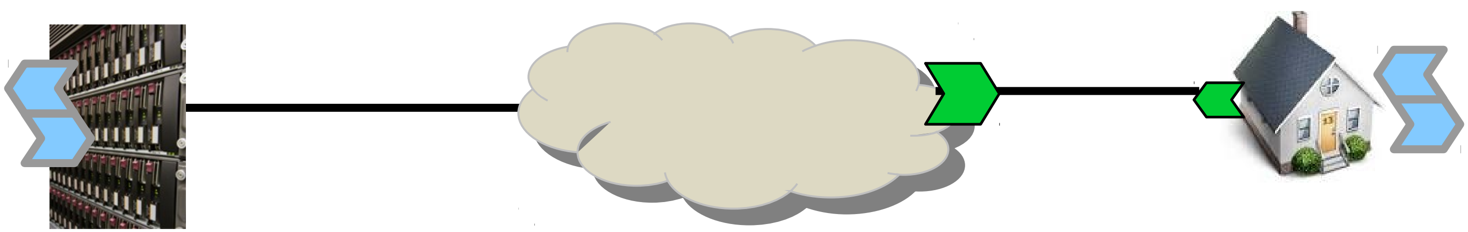
# L4S Deployment

- Coexistence of Classic and L4S traffic
  - DualQ AQM
- Scalable TCP deployment?
  - DCTCP (Data Centre TCP) is a Scalable TCP
  - already in Linux & Windows
  - works over wide area round trip times
  - good enough for controlled trials of L4S
- for production
  - DCTCP needs safety / performance enhancements
  - 'the TCP Prague requirements' [l4s-id]

# L4S Deployment Sequences

## Significant benefit realized at each deployment stage

| | servers or proxies | access link | client |
|---|---|---|---|
| 1. | DCTCP (existing) | DualQ AQM downstream | DCTCP (existing) |
| | works downstream for controlled trials | | |
| 2. | TCP Prague | | AccECN (already in progress – DCTCP/BBR) |
| | works downstream | | |
| 3. | | DualQ AQM upstream | TCP Prague |
| | works upstream & downstream | | |



Where a stage involves 2 moves:

- The benefit after the 2nd move has to be worth the 1st mover's investment risk
- new services or products, not just incremental performance improvement

# Maturity status

- IETF: L4S adopted for standardization (experimental status)
- Numerous companies (often research) involved
  - equipment vendors
  - operators
  - OS developers
  - hardware developers
- Some working on related scenarios
  - e.g. coexistence of TCP and DCTCP in data centres

# further research / open issues

- TCP Prague
  - safety & performance enhancements to DCTCP
- L4S over radio (cellular, WiFi)
  - initial positive results
  - potential to solve TCP's glacial response to radio dynamics
- redesign of rate limiters/policers (e.g. in PoN)
  - currently just use loss – need an ECN-based warning stage
- radio transmission losses
  - watch this space :)

# Further engineering

- Standardization: to be completed
- Network traversal of ECN
  - recent measurements over mobile shows bugs
  - being cleared as (Classic) ECN becomes used
- Implementation of DualQ AQM
  - production software in progress: Linux, NFV (Intel DPDK)
  - compatibility with each vendor's hardware TBA

# more info

https://riteproject.eu/dctth/


- [Briscoe14] Briscoe, B., Brunstrom, A., Petlund, A., Hayes, D., Ros, D., Tsang, I.-J., Gjessing, S., Fairhurst, G., Griwodz, C. & Welzl, M., "Reducing Internet Latency: A Survey of Techniques and their Merits," IEEE Communications Surveys & Tutorials 16(4) IEEE (Nov 2014)
- [RFC7560] Kühlewind, M., Scheffenegger, R. & Briscoe, B. “Problem Statement and Requirements for Increased Accuracy in Explicit Congestion Notification (ECN) Feedback” IETF RFC7560 (2015)
- [Briscoe17] Briscoe, B., Scheffenegger, R. & Kühlewind, M., "More Accurate ECN Feedback in TCP," IETF Internet Draft draft-ietf-tcpm-accurate-ecn-03 (May 2017) (Work in Progress)
- [l4s-id] De Schepper, K., Briscoe (Ed.), B. & Tsang, I.-J., "Identifying Modified Explicit Congestion Notification (ECN) Semantics for Ultra-Low Queuing Delay," Internet Engineering Task Force Internet Draft draft-briscoe-tsvwg-ecn-l4s-id-02 (October 2016) (Work in Progress)
- [dualq-aqm] De Schepper, K., Briscoe (Ed.), B., Bondarenko, O. & Tsang, I.-J., "DualQ Coupled AQM for Low Latency, Low Loss and Scalable Throughput," Internet Engineering Task Force Internet Draft draft-briscoe-tsvwg-aqm-dualq-coupled-00 (October 2016) (Work in Progress)
- [l4s-arch] Briscoe (Ed.), B., De Schepper, K. & Bagnulo, M., "Low Latency, Low Loss, Scalable Throughput (L4S) Internet Service: Architecture," Internet Engineering Task Force Internet Draft draft-ietf-tsvwg-l4s-arch-00 (April 2017) (Work in Progress)

- [PI2] De Schepper, K., Bondarenko, O., Tsang, I.-J. & Briscoe, B., "$PI^2$ : A Linearized AQM for both Classic and Scalable TCP," In: Proc. ACM CoNEXT 2016 pp.105-119 ACM (December 2016)
- [DCttH] De Schepper, K., Bondarenko, O., Tsang, I.-J. & Briscoe, B., "`Data Centre to the Home': Deployable Ultra-Low Queuing Delay for All," (January 2017) (Under Submission)

# Conclusions

- Enables previously infeasible interactive apps
- Need low delay for all of customer's applications at once
  - Differentiated service becoming less useful
- Technical problem: 'Classic' TCP
- Technical solution:
  - "Scalable" TCP with L4S variant of ECN
  - Incremental deployment path
- Business solution:
  - Premium user
  - L4S can become the default Internet service for all users and apps
- Open issues seem to be only around peripheral problems
  - basic solution is ready for take-up

# Q&A

*large saw teeth can ruin the quality of your experience*