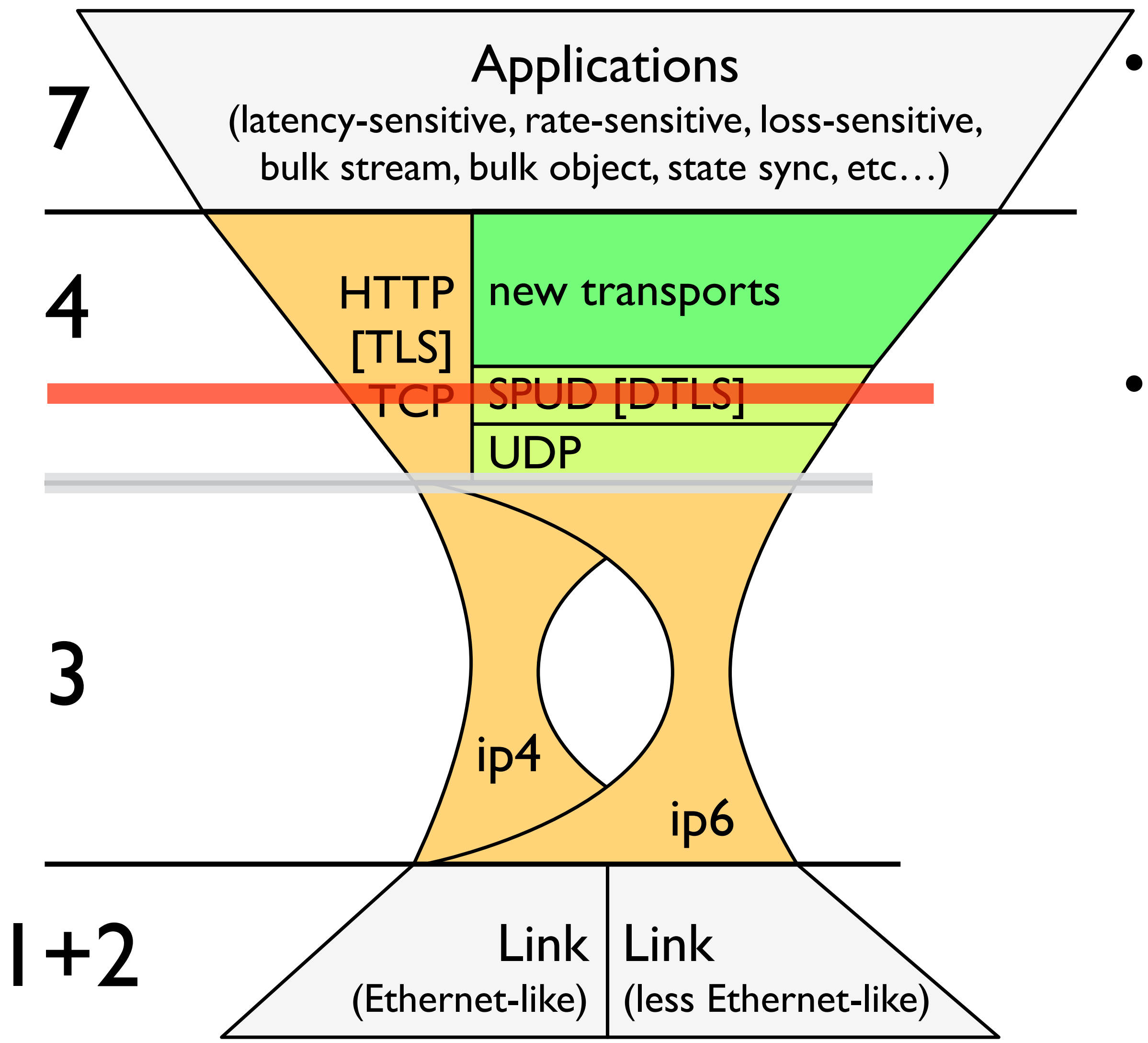
Measurement-Driven Protocol Engineering

IAB Technical Plenary, IETF 94 Yokohama

Brian Trammell, IAB Stack Evolution Program

Evolving the stack:

explicit relayering and cooperation



- Goal: support deployment and user-space experimentation of new transport protocols in today's Internet.
- Approach: Reinforce the layer boundary.
  - UDP encapsulation (ports)
  - crypto (the path only sees what it needs to)
  - explicit cooperation (give back transport and application semantics the path actually needs)

Will any of this actually work?

What can go wrong?

Modification	Planetlab	Ark
NAT	**74.9%**	**79.0%**
ECN IP	**13.7%**	**13.2%**
ISN	**10.7%**	1.8%
MSS	**10.8%**	5.9%
Exp. Option	8.8%	0.5%
MPCAPABLE	8.4%	0.3%
ECN TCP	0.6%	0.6%
SackOK	0.3%	0.0%
TS	0.3%	0.4%
WS	0.2%	0.2%

- NAT everywhere
- Many features mostly work
- Impairments based on implementor and operator assumptions
- Variation based on vantage point
- **Best studies look at O(10k) paths[1].**

[1]: R. Craven, R. Beverly, M. Allman. **A Middlebox-Cooperative TCP for a non End-to-End Internet**. SIGCOMM, August 2014.

How can we do better?

- We want our protocols to work when stuff breaks.
  - Engineering tradeoff: **how many LOC** to avoid a failure that happens **how often**?
  - NAT? Design for it, even if it's hard.
  - A custom hack in one AS squatting on a codepoint? Probably not worth more than the most trivial effort.
- Engineering decisions about protocols to deploy in the Internet should be based on relevant data about the environment they will face.
  - Design for common occurrences.
  - Know the risks of uncommon ones.
  - Apply measurement liberally to know the difference.

Measuring Path Impairment

- Path impairment: the likelihood that traffic with given characteristics will experience problems on a given path.
  - Increased latency, reordering
  - Increased loss/connectivity failure
  - "Bleaching" or selective disablement of features
- Utopian goals:
  - given a proposed protocol feature, know the prevalence of different types of problems with that feature on different networks.
  - given a source and destination, know the types of protocol features that will work along the paths between them.

Measuring the Internet is hard

- Measurements often don't measure what you want.
  - e.g.: ICMP latency and connectivity correlate less than we'd like with application latency and connectivity.
- The Internet is not homogeneous.
  - e.g. how much crypto you see on a given link depends on application mix and the vagaries of CDN policy[2].
- Selection bias: what is easy to measure is not necessarily most relevant.
- Tradeoffs with visibility versus (business) confidentiality and (user) privacy.
- Despite all this, still a lot of insight to gain.

[2]: P. Richter et al. **Distilling the Internet's Application Mix from Packet-Sampled Traffic**. PAM, March 2015.

Improving the best available data

- We have lots of tools, but lack a framework to bring **comparability** and **repeatability** to their observations.
  - Common information models for noting different observations mean similar things.
  - Common measurement control and query protocols[3].
- Opportunities to **glean** data from non-measurement activities.
  - TCP measures itself as a side-effect of its operation.
  - Platform-level diagnostics[4] a potentially rich source.
  - Future: measurement as an explicit protocol feature.
- First step: sharing what we know. **HOPSRG, [date/time]**

[3] e.g. mPlane, ict-mplane.eu, draft-trammell-mplane-protocol

[4] e.g. telemetry.mozilla.org

A principle of measurement

- There are many other insights to be gained from the Internet by measuring it in different ways.
- Can this approach be generalized beyond this particular problem?
- What sorts of insight can be gained from [area of measurement]?