Internet capacity sharing: a way forward?

Bob Briscoe
Chief Researcher, BT

Jul 2009

This work is partly funded by Trilogy, a research project supported by the European Community
www.trilogy-project.org

Internet capacity sharing – a huge responsibility

- getting this right will free up a huge variety of source behaviours
  - ‘TCP-friendly’ has limited our imaginations
  - TCP’s rate response to congestion is sound (still important)
  - but endpoint algos alone cannot be the basis of capacity sharing
- getting it wrong leaves ISPs no choice but to close off the future
  - ISPs resort to app analysis (deep packet inspection)
  - getting impossible to deploy a new use of the Internet
  - must negotiate the arbitrary blocks and throttles en route
- design team’s premise
  - capacity sharing function belongs primarily to the network
  - what’s a minimal network function? which preclude future options?
- grudging acceptance of proverb: "good fences make good neighbours"
  - not natural for most of us to design fences
  - but lacking a good fence design, the industry is building bad ones
    - cf. lack of a place for firewalls and NATs in IETF/IRTF architecture

Internet capacity sharing architecture

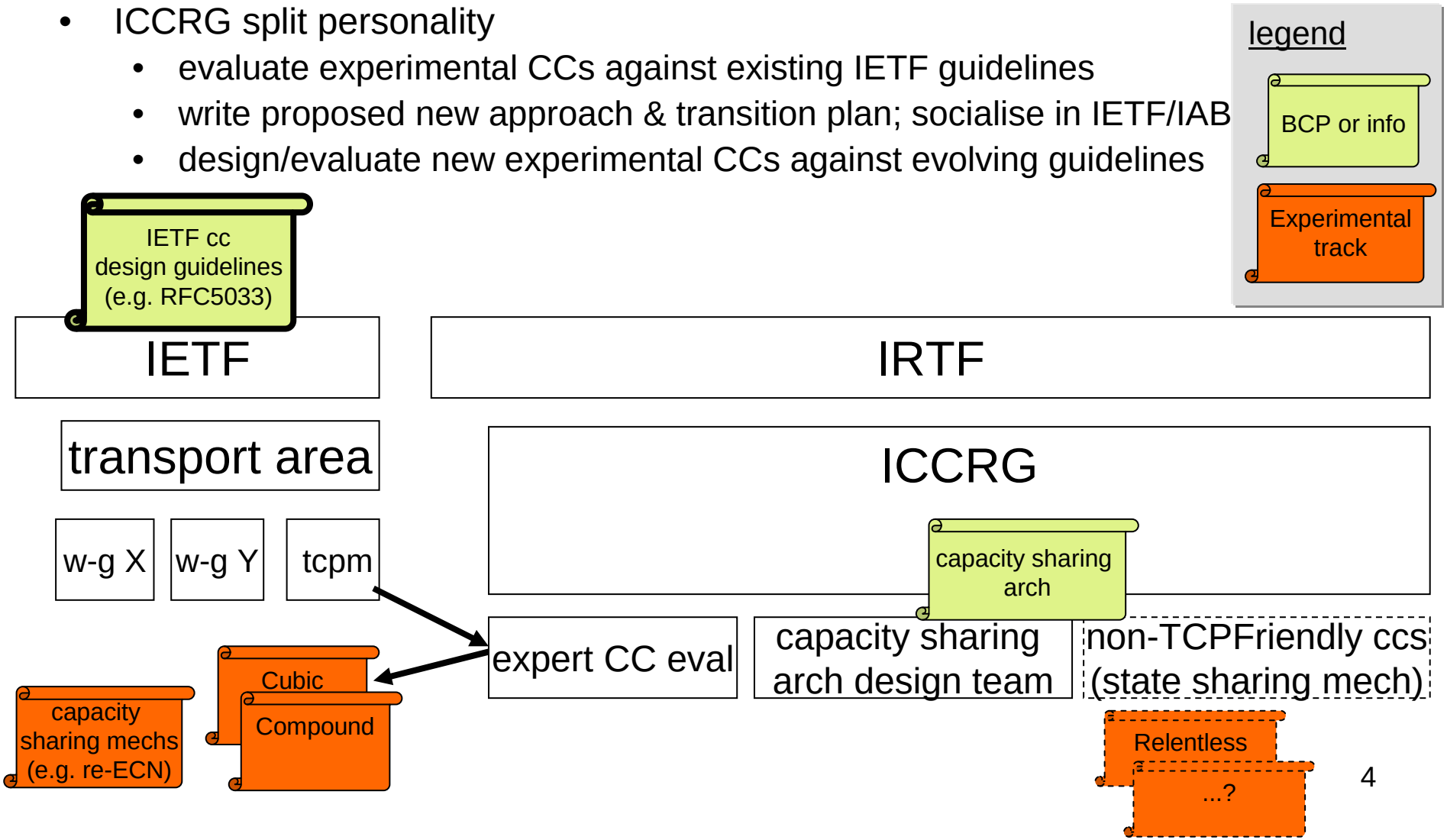
design team status

- goal
  - informational RFC recording IRTF consensus on how to shift to a new capacity sharing architecture for the Internet
  - input to possible subsequent IAB & IESG consensus
- modus operandi
  - touch consensus forming task
  - team works off-list, progress & review on iccrg list
  - http://trac.tools.ietf.org/group/irtf/trac/wiki/CapacitySharingArch
- people
  - by incremental invitation; not too large
  - need different worldviews but some common ground
  - Matt Mathis, Bob Briscoe, Michael Welzl, Mark Handley, Gorry Fairhurst, Hannes Tschofenig, ...

Internet capacity sharing architecture; design team

relation to other ICCRG/IETF activities

- ICCRG split personality
  - evaluate experimental CCs against existing IETF guidelines
  - write proposed new approach & transition plan; socialise in IETF/IAB
  - design/evaluate new experimental CCs against evolving guidelines



history of capacity sharing goals

- consensus growing that TCP-friendly is not the way forward
  - recurrent goal since at least mid-1970s: competing flows get equal bottleneck capacity
  - 1985: fair queuing (FQ): divide capacity equally between source hosts
    - limited scope recognised: per switch & src addr spoofing
  - 1987: Van Jacobson TCP, window fairness
    - limited scope recognised: hard to enforce
  - 1997: TCP friendliness: similar average rate to TCP, but less responsive. Increasingly IETF gold standard
  - 1997: Kelly weighted proportional fairness optimises value over Internet based under congestion pricing
  - 2006: Briscoe capacity sharing is about packet level, not flow level
- Nov 2008: Beyond TCP-friendly design team in IRTF created, following consultation across IETF transport area
- Mar 2009: Non-binding straw poll in IETF transport area: no-one considered TCP-Friendly a way forward
- May 2009: two ICCRG CC evaluation strands for capacity sharing:
  - TCP-friendly for present IETF
  - network-based (TBD) for new CCs

design team's top level research agenda?

- statement of ultimate target
  - metrics & deprecated metrics
  - structure & deprecated structure
  - enduring concepts
- standards agenda
  - 1/p congestion controls
  - weighted congestion controls
  - congestion transparency (re-ECN)
- deployment scenarios
  - unilateral
  - co-ordinated

metrics

i	flow index
x	bit-rate
p	marking fraction

- deprecated metrics
  - hi-speed flows competing with low is perfectly ok
  - relative flow sizes at a resource not relevant to fairness
  - blocking exceptionally high flow rates deprecated
- competition with legacy
  - s/equal windows within an order of magnitude
    /avoid legacy flow starvation & ratchet down effects/
  - shift from relative rates to sufficient absolute legacy rate
- ultimate target metrics
  - congestion-volume $\equiv \Sigma_i \int p(t)x_i(t)\,dt$

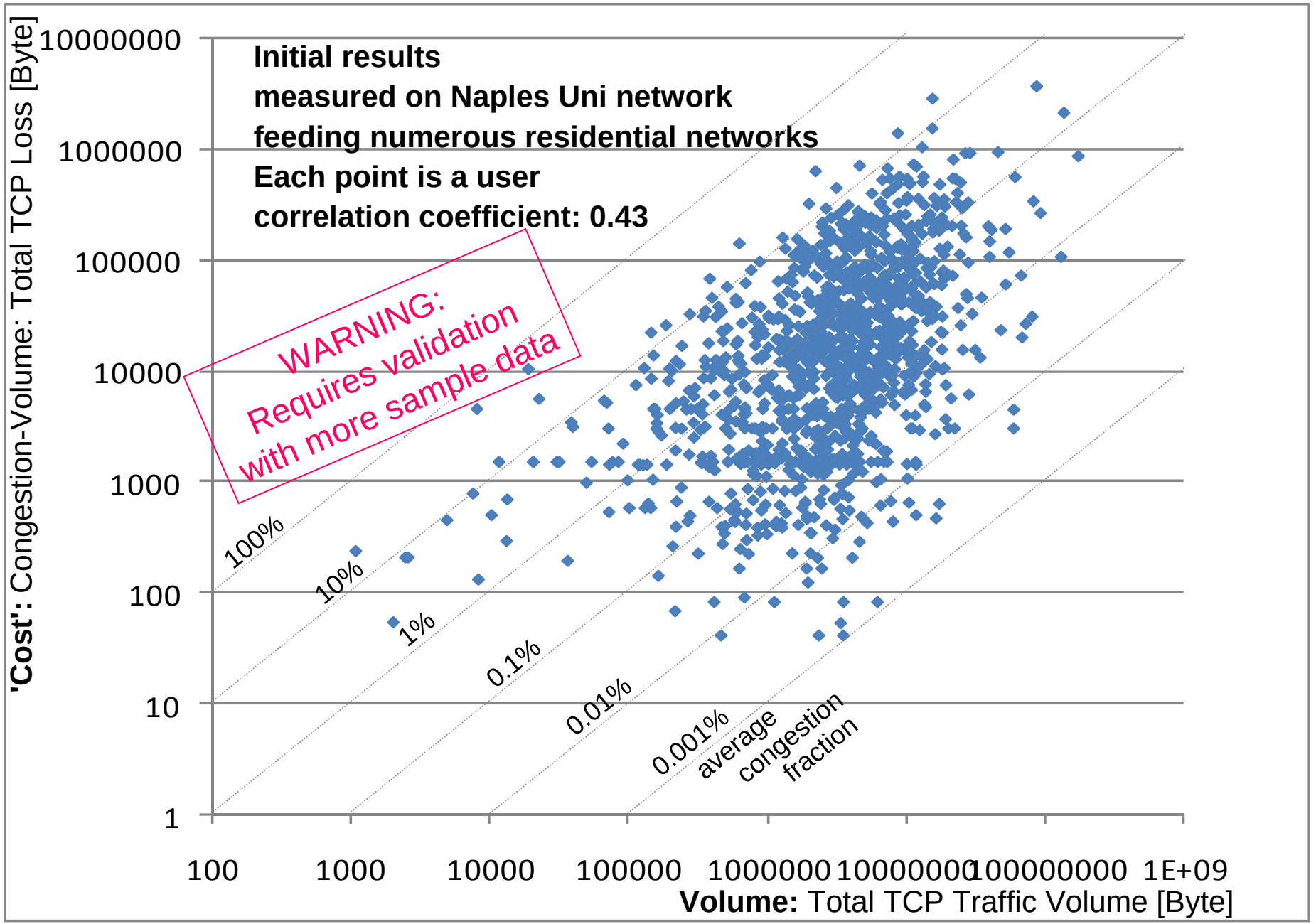
    volume of marked bits != volume $\equiv \Sigma_i \int x_i(t)\,dt$
  - congestion-bit-rate $\equiv \Sigma_i\, p(t)x_i(t)$

    rate of lost / marked bits; != aggr. bit-rate $\equiv \Sigma_i \; x_i(t)$

metrics

per-flow bit-rate policing *deprecated!!?*

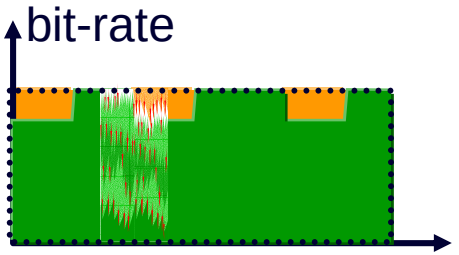
- per-flow bit-rate policing != per-user bit-rate policing
  - ultimately share access networks by congestion-bit-rate
  - as interim, per-user rate policing doesn't close off much
    - just as if a shared link were multiple separate links
  - but per-flow rate policing closes off a lot of future flexibility
    - and it's unnecessary to satisfy anyone's interests
- i.e. WFQ on access link is fairly harmless as interim
  - still not ideal for resource pooling
    - prevents me helping you with LEDBAT
      - I can only help myself
    - isolation between users also isolates me from other users' congestion signals
    - can't respond even though I would be willing to


Initial results
measured on Naples Uni network
feeding numerous residential networks
Each point is a user
correlation coefficient: 0.43
WARNING:
Requires validation
with more sample data
'Cost': Congestion-Volume: Total TCP Loss [Byte]
10000000
1000000
100000
10000
1000
100
10
1
100%
10%
1%
0.1%
0.01%
0.001%
average
congestion
fraction
100
1000
10000
100000
1000000
10000000
100000000
1E+09
Volume: Total TCP Traffic Volume [Byte]

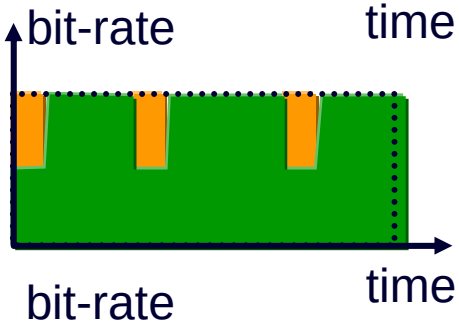
motivating congestion-volume

weighted congestion controls

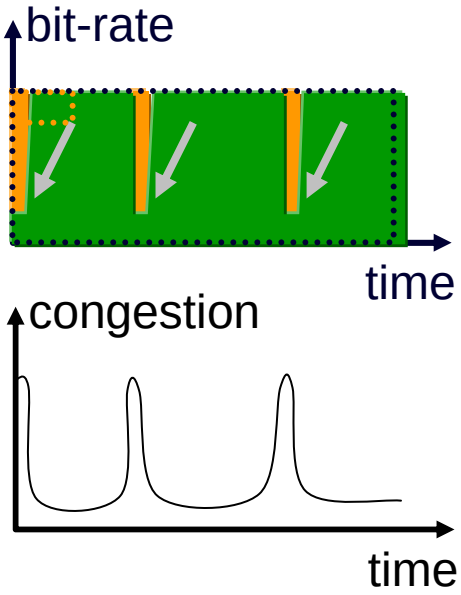
1. TCP



2. WFQ



weighted sharing



- light usage can go much faster
- hardly affects completion time of heavy usage

NOTE: weighted sharing doesn't imply differentiated network service

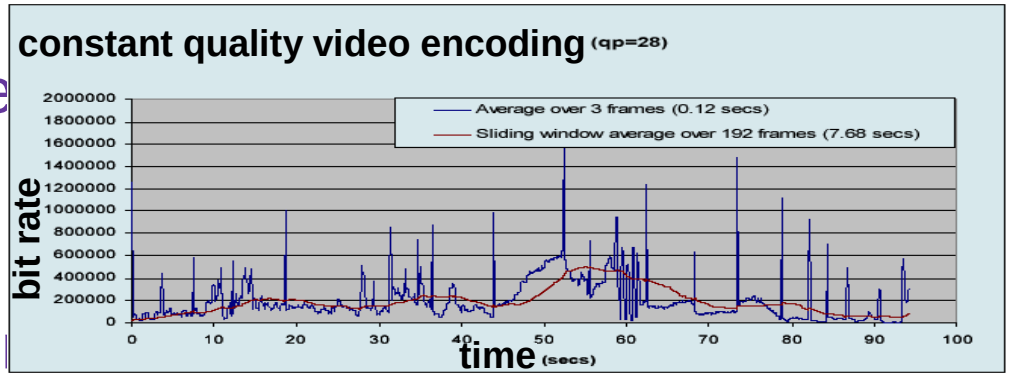
- just weighted aggressiveness of end-system's rate response to congestion
- LEDBAT: a fixed weight example

motivating congestion- volume

harnessing flexibility

guaranteed bit- rate?

or much faster 99.9% of the ti



- the idea that humans want to have a known fixed bit-rate
  - comes from the needs of media delivery technology
  - hardly ever a human need or desire

- services want freedom & flexibility
  - access to a large shared pool, not a pipe
- when freedoms collide, congestion results
  - many services can adapt to congestion
  - shift around resource pool in time/space

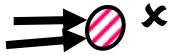
% figures = no. of videos that fit into the same capacity

Constant Bit Rate **100%** Constant Oualitv **125%** Equitable Quality **216%**

sequences encoded at same average of 500kb/s [Crabtree09]

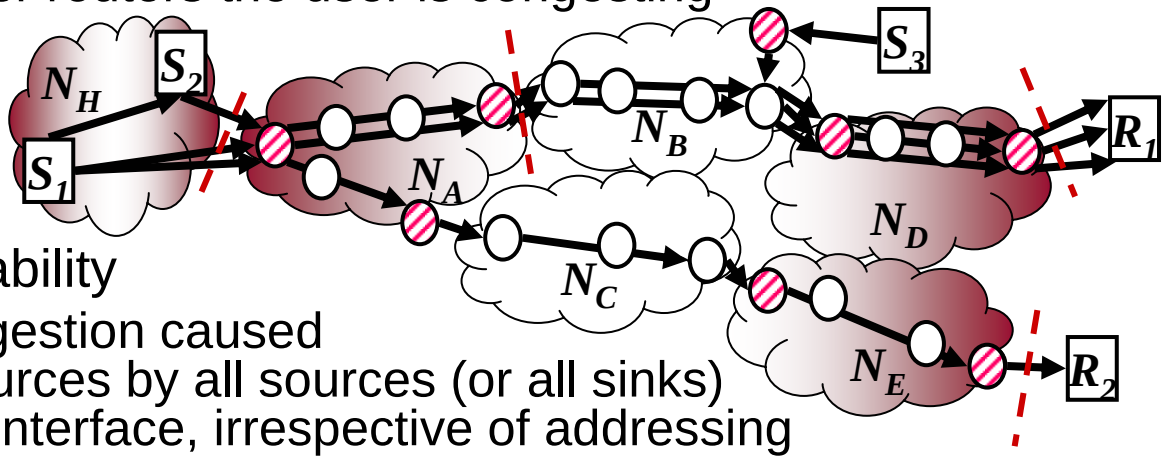
target structure: *network* fairness

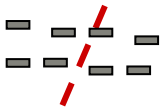
difference is clearest if we consider enforcement structures



× bottleneck policers: active research area since 1999

- detect flows causing unequal share of congestion
- located at each potentially congested router
- takes no account of how active a source is over time
- nor how many other routers the user is congesting
- based on cheap pseudonyms (flow IDs)





✓ congestion accountability

- need to know congestion caused in all Internet resources by all sources (or all sinks) behind a physical interface, irrespective of addressing
- no advantage to split IDs
- each forwarding node cannot know what is fair
- only contributes to congestion information in packets
- accumulates over time
- like counting volume, but ‘congestion-volume’

- focus of fairness moves from flows to packets

enduring concepts, but nuanced

- end-point congestion control (rate response)
  - with weights added
    & network encourages weights to be set sparingly
- random congestion signals (drops or marks) from FIFO queues
  - marks preferred – network can't measure whole-path drop
  - holy grail if feasible – new cc with old AQM?
  - has to work well enough, optimisation can be piecemeal
- Diffserv?
  - less than best effort scheduling
  - may be necessary for incremental deployment
  - may be necessary in long term?
- Diffserv & congestion signals: point of current debate

design team's top level research agenda?

- statement of ultimate target
  - metrics & deprecated metrics
  - structure & deprecated structure
  - enduring concepts
- standards agenda
  - 1/p congestion controls
  - weighted congestion controls
  - congestion transparency (re-ECN)
- deployment scenarios
  - unilateral
  - co-ordinated

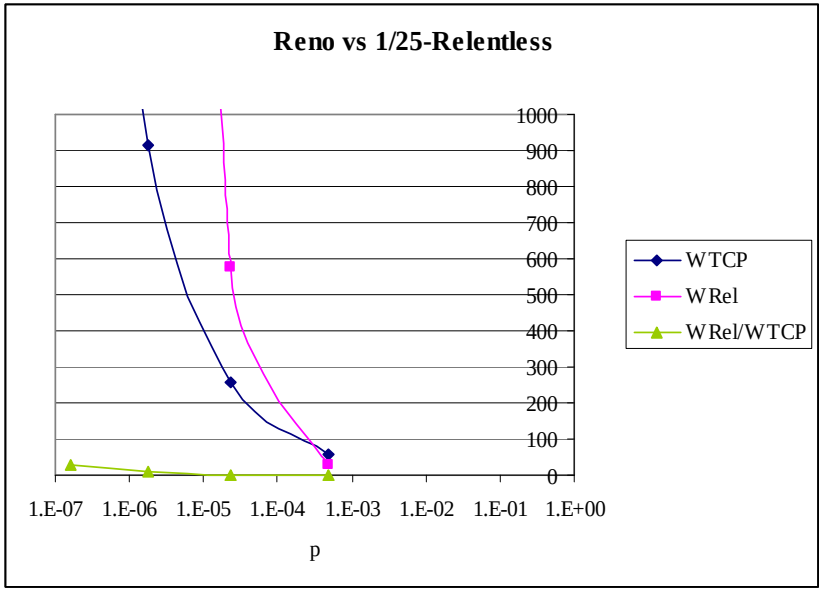
a basis for consensus?

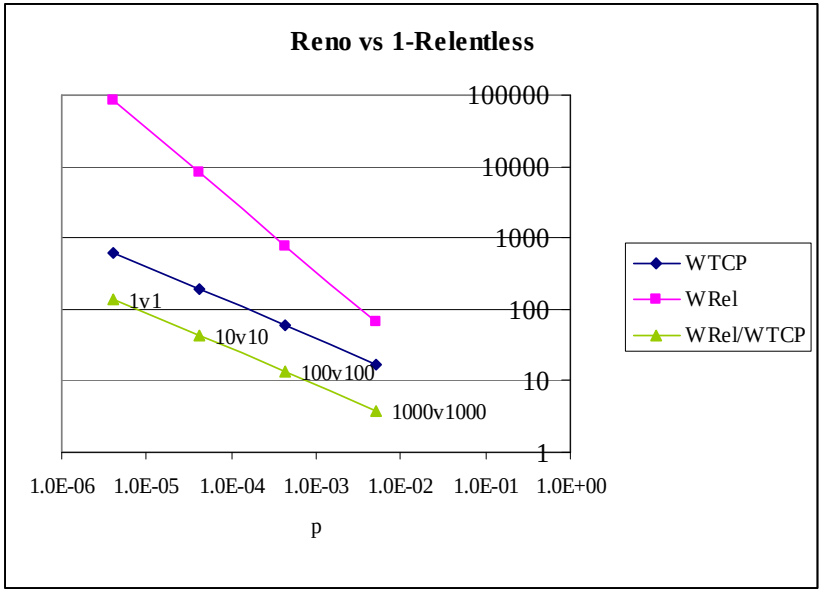
1/p congestion controls (e.g. Relentless CC)

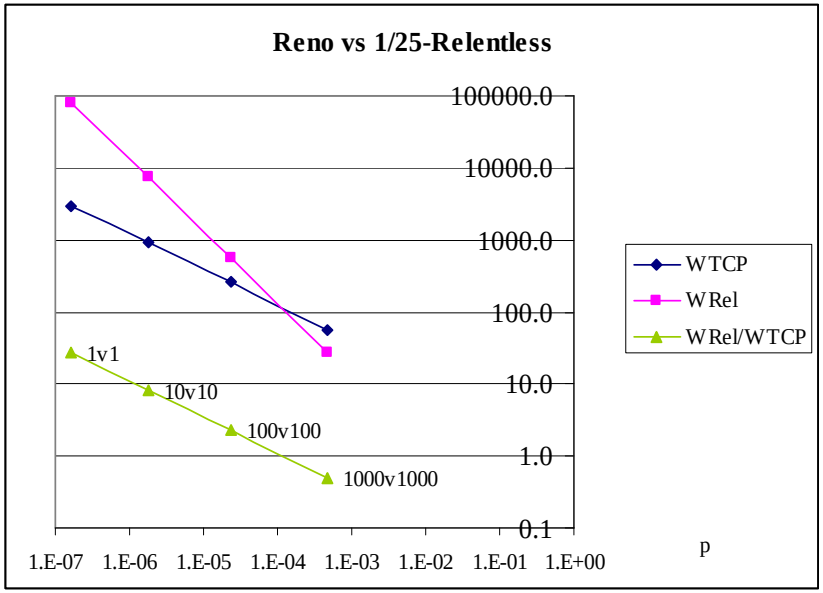
- TCP's $W \propto 1/\sqrt{p}$ window doesn't scale
  - congestion signals /window reduce as speed grows, $O(1/W)$
  - root cause of TCP taking hours / saw tooth at hi-speed
- $W \propto 1/p$ scales congestion signals / window $O(1)$
  - Relentless, Kelly's primal algorithm
  - IOW, get same no of losses per window whatever the rate
- an alternative way of getting more precise congestion signals than more bits per packet

weighted congestion controls

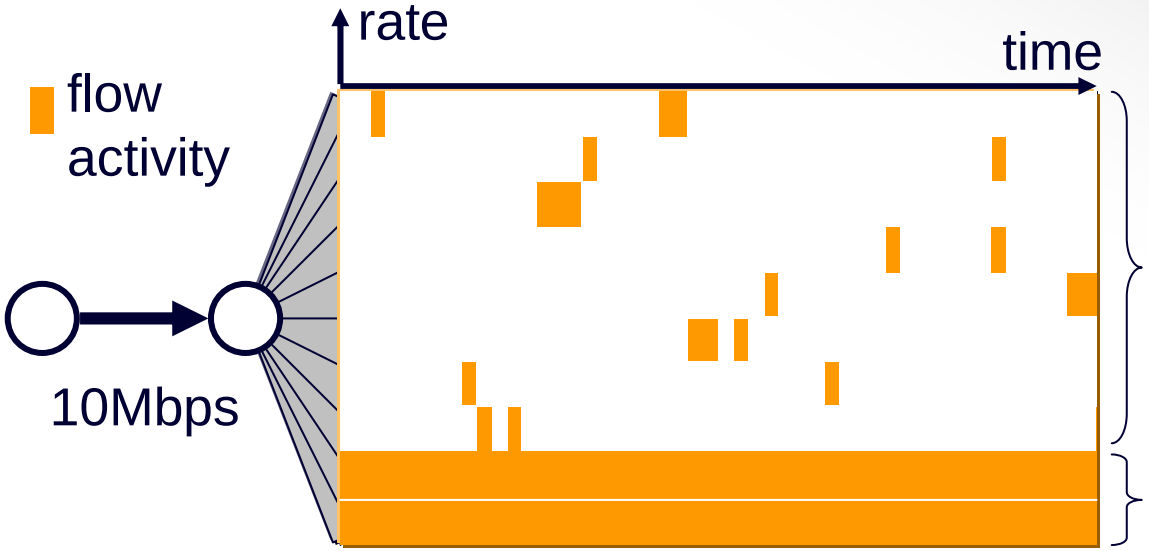
- toy models
  - don't fret over numbers
  - p: loss/marking fraction (log scale)
- weighted w-Relentless TCP ($w=^{1}/_{25}$)
  - on every mark/loss W –= 25
  - just FIFO queues
- Reno gets 'enough' over range
  - would hardly do better alone
  - if it's not enough, upgrade







Reno vs. w-Relentless
no less flow starvation than TCP-friendly



2Mbps access each

80 users of attended apps

20 users of unattended apps

usage type	no. of users	activity factor	ave.simul flows /user	TCP bit rate /user	vol/day (16hr) /user	traffic intensity /user
attended	80	5%	=	417kbps	150MB	21kbps
unattended	20	100%	=	417kbps	3000MB	417kbps
				x1	x20	x20

standards agenda

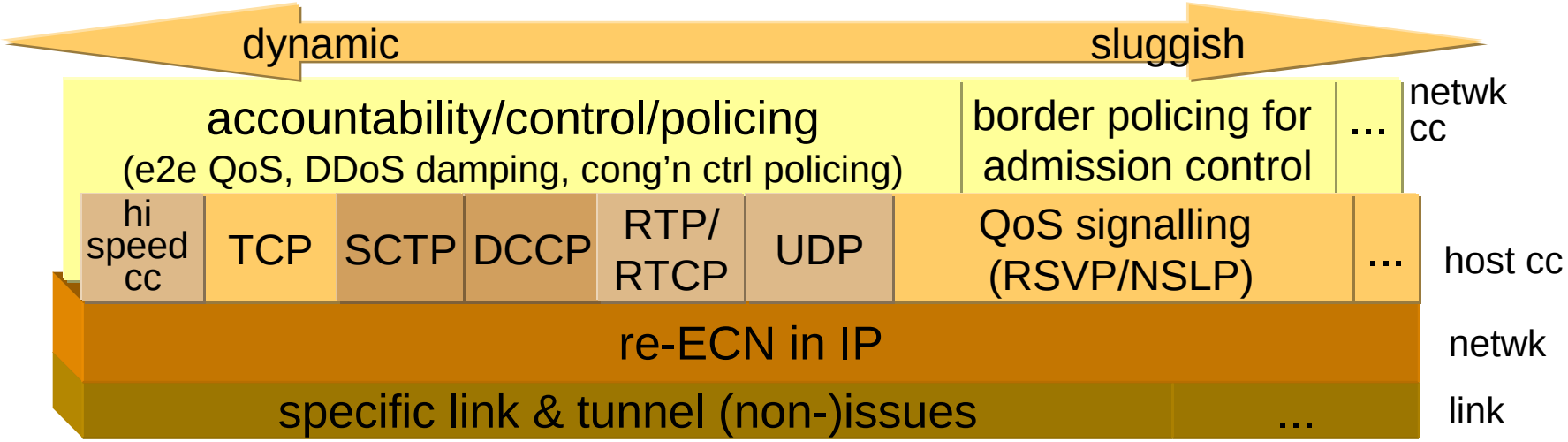
weighted congestion controls

- important to enable $w<1$, negates weight inflation
- add weight to all(?) new congestion controls
  - LEDBAT, mTCP, SCTP, Relentless ...
- new app parameter overloading socket API
  - also app & policy integration
- timing relative to ability to police is tricky
  - change to IP will take much longer than new cc algos
  - perhaps have weighting in cc algo,
    but hard-code a value without an API until later

standards agenda

re-ECN

- source reveals congestion to net in IP header
- work to get to standards track
  - re-ECN in IPv6
  - re-ECN in IPv4 (experimental)
    - in controlled environments (e.g. GENI slice)
  - re-ECN in various transports
  - tunnelling IPv6 re-ECN in IPv4?



- the work that will take longest ought to finish first
  - Transport Area, Network Area, Security Area, etc.
  - should we take a punt before agreeing the way forward
    - Congestion Transparency (re-ECN) BoF in Stockholm?

congestion transparency (re-ECN) bar BoF

- Thu 15:10- 16:10 Rm 501
- Not slides about re ECN
- getting together people interested in getting a BoF together at future IETF
  - experimental protocol

a vision: flat fee congestion policing

if ingress net could see congestion...

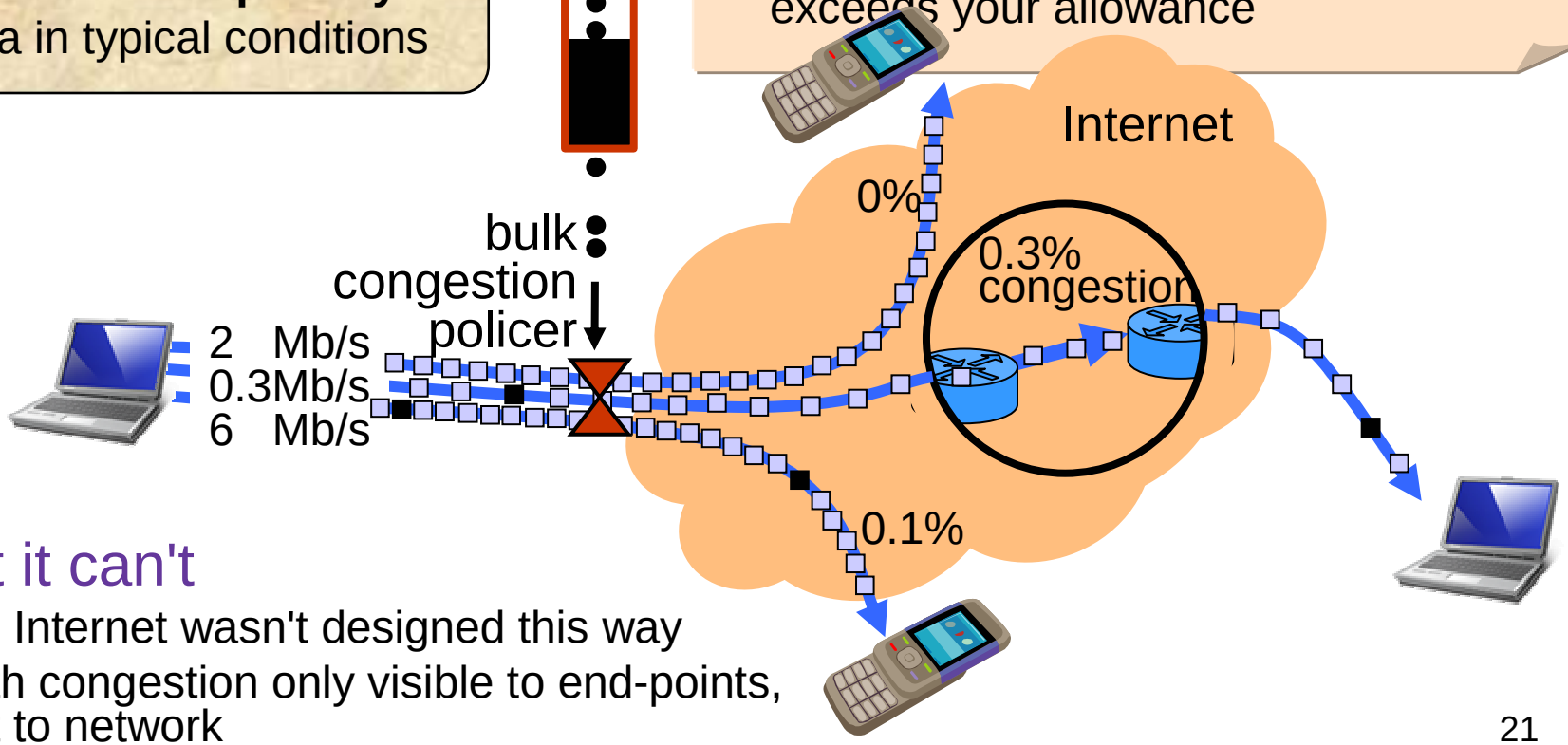
Acceptable Use Policy

'congestion-volume' allowance: 1GB/month

@ £15/month

Allows ~**70GB per day** of data in typical conditions

- incentive to avoid congestion
- simple invisible QoS mechanism
  - apps that need more, just go faster
- side-effect: stops denial of service
- only throttles traffic when your contribution to congestion in the cloud exceeds your allowance



...but it can't

- the Internet wasn't designed this way
- path congestion only visible to end-points, not to network

design team's top level research agenda

- statement of ultimate target
  - metrics & deprecated metrics
  - structure & deprecated structure
  - enduring concepts
- standards agenda
  - 1/p congestion controls
  - weighted congestion controls
  - congestion transparency (re-ECN)
- deployment scenarios
  - unilateral
  - co-ordinated

a basis for consensus?

deployment scenarios

assumption space of in-network mechanisms

- hi/med/lo statistical multiplexing
- LE (less than best effort Diffserv)
- AQM
  - ECN
  - ECN across Diffserv queues, vs separate
    - virtual queues

- work in progress, mapping out this space
  - which of these are necessary?
  - what happens when not all routers support them?
  - does each only matter in certain stat mux cases?

is the Internet moving to multiple bottlenecks?

- receive buffer bottleneck likely cause of lack of congestion in cores
- window scaling blockages are disappearing
- machines on campus & enterprise networks (not limited by access bottlenecks) will increasingly cause bursts of congestion in network cores
- removes old single bottleneck assumptions
  - complicates capacity sharing deployment
  - e.g. WFQ has been used in access networks
    - by assuming single bottleneck
    - CSFQ (core state fair queuing) extends FQ
    - but (CS)FQ doesn't help resource pooling (see earlier)

unilateral deployment scenario example
(non-TCP-friendly, ECN, re-ECN)

- no congestion transparency (not in protocols)
  - operator uses local congestion-volume metric in place of volume at single bottleneck (e.g. on traffic control boxes)
  - end-host acts as if congestion-volume is limited
  - appears as voluntary as TCP, but unlikely to happen?
  - cf. BitTorrent, Microsoft & LEDBAT

more info

Re-architecting the Internet:

The Trilogy project <www.trilogy-project.org>

re-ECN & re-feedback project page:

http://www.cs.ucl.ac.uk/staff/B.Briscoe/projects/refb/

These slides

<www.cs.ucl.ac.uk/staff/B.Briscoe/present.html>

bob.briscoe@bt.com

deployment incentives

[re-ECN06] *Using Self-interest to Prevent Malice; Fixing the Denial of Service Flaw of the Internet*, Bob Briscoe (BT & UCL), The Workshop on the Economics of Securing the Information Infrastructure (Oct 2006)

[re-ECN] <draft-briscoe-tsvwg-re-ecn-tcp>

[re-ECN09] <draft-briscoe-tsvwg-re-ecn-tcp-motivation>

[Crabtree09] B. Crabtree, M. Nilsson, P. Mulroy and S. Appleby “Equitable quality video streaming” Computer Communications and Networking Conference, Las Vegas, (Jan 2009)

ECN @ L2

[Siris02] ``Resource Control for Elastic Traffic in CDMA Networks'' In *Proc. ACM MOBICOM 2002*, Atlanta, USA, 23-28 (2002). <www.ics.forth.gr/netlab/wireless.html>

ECN @ L4-7

[RTP-ECN] draft-carlberg-avt-rtp-ecn

[RTCP-ECN] draft-carlberg-avt-rtcp-xr-ecn

Internet resource sharing: a way forward?

discuss...