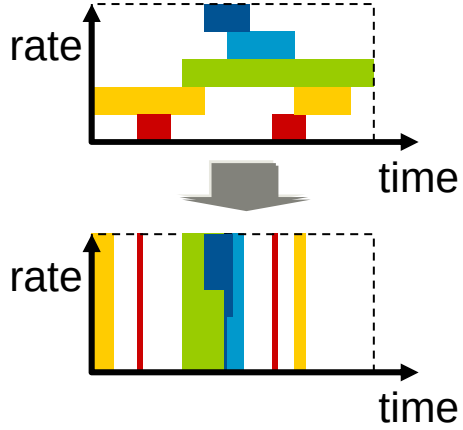
nice traffic management
without new protocols

Bob Briscoe
Chief Researcher, BT

Oct 2012

vertical stripes: this season's colour



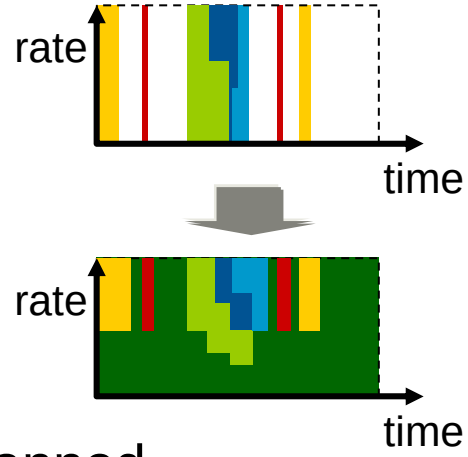
viewed at second mile

- increasingly access no longer the bottleneck
  - PON, FTTP, FTTdp, FTTC
  - bottleneck moving deeper
  - becoming similar to campus LAN, data centre
- each customer's average access utilisation v low
  - 1-3% average during peak hour (some 100%, but rarely at the same time)
  - if provision bottleneck for the worst case load seen, it leaves a lot of leeway for much worse cases
- traditionally the bottleneck solves this with
  - (weighted) fair-queuing / round robin*

- about isolation from random bad performance
  - not about skimping on capacity

* or policed per-customer b/w limits (for higher utilisation customers e.g. DC)

fair queuing: so 1990s



- enforces 1/N shares* so that's fine?
- No
  - when average N is so low
  - a few more long-running customers than planned can increase N significantly
  - thereby greatly decreasing everyone else's 1/N share
- the problem:
  - N depends heavily on presence of high util'n customers
  - usually few, but when many, service seems crap
  - large buffers make this much worse but not the only problem
- 1/N is 'fair' at each instant, but not over time

* as does WRR (and TCP sort-of)

ConEx: so y=2014

y = y(now) + 2

the ConEx rationale

- FQ + volume limits?
  - hi vol customer only a problem if with other hi vol cust's
- Lack of complete solution led to non-neutral solutions, then...
- Comcast fair-share
  - limit highest volume customer(s) only if cable congested
  - better, but penalises hi-vol even if transport yields [LEDBAT]
- research goal: we ain't seen nothing yet on the Internet
  ...if we designed the network for un-lame transports

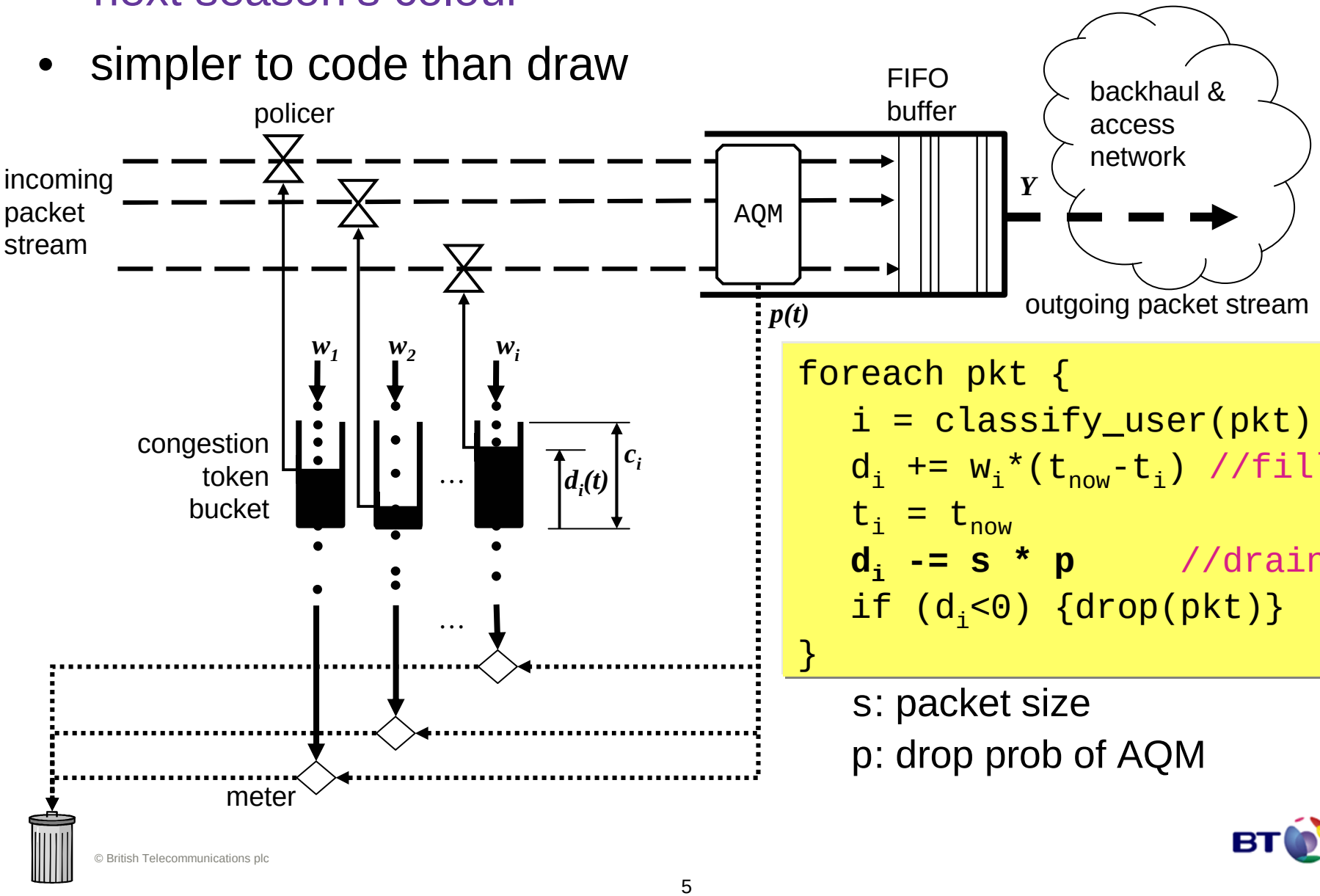
- ConEx rationale is actually in two parts:
  1. rationale for using congestion-volume as a metric
  2. need a change to IP (ConEx) to see congestion-volume

Is there an 80% solution without changing IP?

BT

bottleneck congestion policer:
next season's colour

- simpler to code than draw



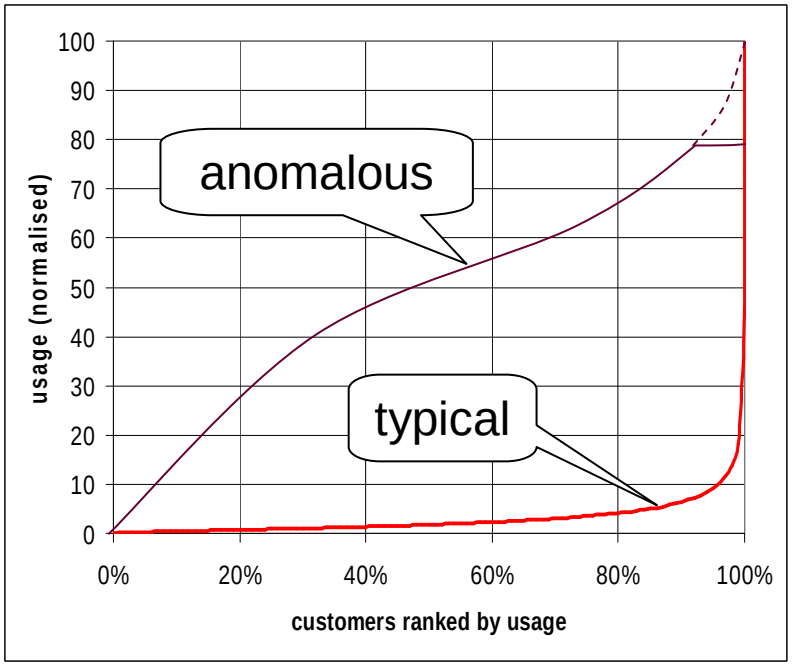

```
foreach pkt {
   i = classify_user(pkt)
   d_i += w_i*(t_now-t_i) //fill
   t_i = t_now
   d_i -= s * p      //drain
   if (d_i<0) {drop(pkt)}
}
```

s: packet size

p: drop prob of AQM

bottleneck congestion policer (BCP): features

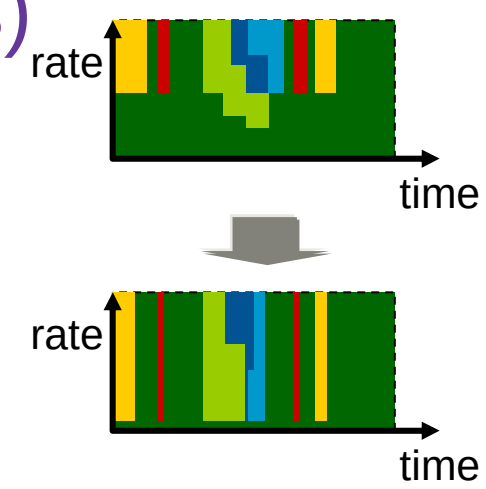
- predictable quality for the many
  - keeps queue v short
  - by focusing discard on those who most push at the queue*
- tends to WRR if customer traffic becomes continuous
- app-neutral



- applicability
  - same as any per-customer limiter
  - state: per-customer configured allowance and usage-level
  - drop-in to: BRAS / MSE, RNC, OLT, DC access
  - few simultaneous customers (or many)
- where bottleneck location varies, still need to evolve to ConEx

* relative to their allowance to do so

next steps (next season's shoes)



- plan to open-source BCP
  - not yet fully agreed internally
- baby step
  - gets industry used to congestion-volume as the metric

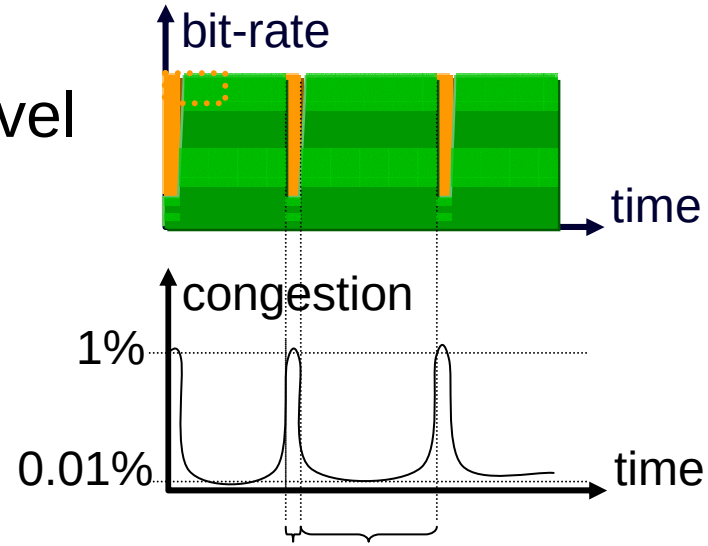
nice traffic management
without new protocols

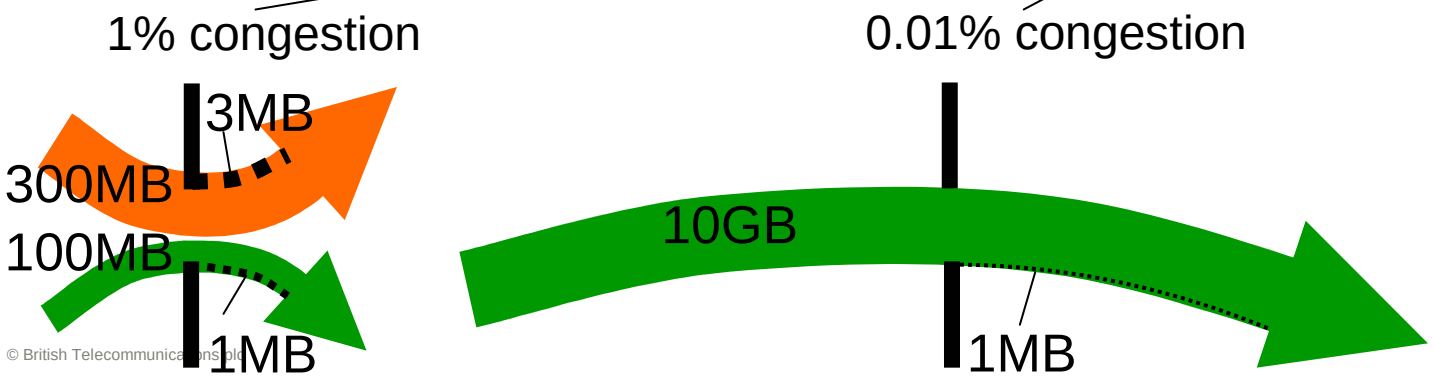
Q&A
discussion
spare slide

measuring contribution to congestion

= bytes weighted by congestion level
= bytes dropped (or ECN-marked)
= ‘congestion-volume’
∝ marginal cost of capacity

- as simple to measure as volume





actually each bucket needs to be two buckets to limit bursts of congestion