Latency in DOCSIS Networks

CableLabs®

Greg White
Sept. 26, 2013

The various DOCSIS versions

From a latency perspective

- DOCSIS 1.0 – ca. 1996, deployments ~1998
  - Fundamental request-grant upstream MAC layer definition
- DOCSIS 1.1 – ca. 1999, deployments ~2001
  - Additions for configured Quality of Service
    - Packet classifiers
    - Flow independence
    - QoS configuration per flow
- DOCSIS 2.0 – ca. 2001, deployments ~2003
  - TCP ACK suppression/prioritization
- DOCSIS 3.0 – ca. 2006, deployments ~ 2007
  - Buffer Control feature added in 2011
- DOCSIS 3.1 – ca. 2013, deployments ~2015
  - Light Sleep Mode
  - AQM Mandatory

CableLabs®

DOCSIS Request-Grant Upstream MAC

- Upstream channel scheduling is driven by “MAP” Intervals (typ. 2ms)

- Packet(s) arrive at the cable modem (CM)
- CM waits* for the next contention request opportunity
  - *typically less than 2ms
- CM sends request message (subject to rate shaping)
- CMTS** scheduler collects requests, then schedules and communicates future transmit opportunities (grants)
- Due to serialization, propagation and interleaver delays, as well as CMTS/CM processing delays, grant occurs 2 MAP Intervals after the request was sent

- Without congestion, typically 4-8ms access latency

**Cable Modem Termination System

CableLabs®

Quality of Service (D1.1 and above)

- For known applications with known QoS requirements
- Operator configures packet classifiers and service flow QoS parameters
  - Token bucket rate shaping, priority, guaranteed rate, low-latency scheduling, etc.
- Service Flows queue traffic and access channel independently
- Modems today support 16 or 32 service flows, each with an independent hardware queue

CableLabs®

TCP ACK Suppression/Prioritization

Dealing with buffer bloat before "Bufferbloat"

- Queue build-up from upstream TCP sessions delays upstream TCP ACKs – downstream throughput suffers.
- All modem vendors implement proprietary mechanisms in D2.0 and above to move ACKs to the head of the queue and discard superfluous ACKs.
- TCP RTT depends on which side you measure from

CableLabs®

Buffer Control (D3.0)

Bufferbloat is everywhere!

- Amended specification in 2011 to allow operator to set per-service flow buffer sizes.

- Requires configuration by operator
- Interest was high, adoption has been slow.

CableLabs®

Light Sleep Mode (D3.1)

Reducing Energy Consumption during "Idle"

- Response to Political Pressures such as:
  - **More Efficient Modems, Routers Could Save Consumers $330 Million Annually – NRDC**
    - "These small, **innocuous black boxes** that never sleep consume enough electricity each year to power all 1.2 million homes in the Silicon Valley area, the hi-tech capital of the world," said NRDC senior scientist Noah Horowitz. "Small network devices suck roughly the same amount of energy **around the clock**, whether or not you are sending or receiving any data. But there are steps that manufacturers can – and should – take to make sure these devices are no longer **energy vampires**."
    - 88 million Internet consumers in US:
      - **More Efficient Modems, Routers Could Save Each Consumer $0.31 Monthly**
  - EPA Energy Star - Small Network Equipment Spec
  - California Energy Commission – Consumer Electronics Efficiency Pre-Rulemaking
  - EU Lot 26 & Networked Standby Regulation
  - EU Broadband Code of Conduct

CableLabs®

Light Sleep Mode (D3.1)

- Modems in Light Sleep Mode will shut down receiver for periods of up to 200ms.
  - Interval set by CMTS, could be less.
  - Downstream packets queued at CMTS until wake interval
  - Baseline latency measured during network idle conditions may not give you the results you expect.
  - Latency under load might actually be better than "baseline"

CableLabs®

Active Queue Management (D3.1)

- AQM will be mandatory for both CM and CMTS in D3.1
- On by default, can be disabled on a per service flow basis
- CMTS can implement an algo. of the vendor’s choosing
- CM MUST implement single-queue PIE, but can also implement other algorithms
  - PIE chosen over CoDel, CoDel-DT, SFQ-CoDel, SFQ-PIE
- Currently investigating if existing D3.0 equipment can be upgraded to support AQM

CableLabs®

Why no *FQ?

- Hardware complexity of 32 Service Flows x 32 queues
  - Or, operational complexity of Service Flows sharing a pool of N queues
- Tight deadlines between MAP & grant
  - *any* additional processing at dequeue time is hard
- Limited additional benefit compared to single queue AQM at 100Mbps+
- Concerns about VPN traffic
- Hash collisions – not feasible to have 1024 queues (see above)

CableLabs®