Impact of Topology on Layer 2 Switched QoS Sensitive Services

Bart Puype, Geert Verbanck*, Jurgen Michielsen*, Marc Moeskops*,
Wouter Tavernier, Didier Colle, Mario Pickavet, Piet Demeester
Department of Information Technology (INTEC – IBCN), Ghent University – IBBT,
Gaston Crommenlaan 8 bus 201, 9050 Gent, Belgium
**OTN Systems NV, Atealaan 34, 2200 Herentals, Belgium*
Tel: +32 9 33 14900, Fax: +32 9 33 14899, e-mail: bart.puype@intec.ugent.be

ABSTRACT

High-bandwidth QoS sensitive services such as large scale video surveillance generally depend on provisioned capacity delivered by circuit-switched technology such as SONET/SDH. Yet development in layer 2 protocol sets and manageability extensions to Ethernet standards propose layer 2 packet switching technology as a viable, cheaper alternative to SONET/SDH. Layer 2 switched networks traditionally offer more complex topologies; in this paper we explain general QoS issues with layer 2 switching and show the impact of topology choice on service performance.

Keywords: layer 2 switching, quality of service, topology.

1. INTRODUCTION

Electronic circuit-switching technology has existed for quite some time, and was originally envisioned for telephony services. Circuit-switching has culminated into SONET/SDH technology, which offers a wide range of features that facilitate manageability, resilience and quality of service. As bandwidth needs grew larger, optical transmission was used in the physical layer. At the same time, required bandwidth of services grew larger as well. Of course one such service consists of best-effort TCP/IP packet transmission for internet connectivity. More interestingly, digital transmission also became popular for services that traditionally were carried over an analog encoding. Given the guaranteed high bandwidth needs and QoS constraints of such services such as video streaming, SONET/SDH is the obvious digital transmission technology.

On the other hand, the packet switched TCP/IP layer model defines a packet (or frame) switched transport layer (layer 2). Ethernet and related IEEE defined physical layers are the de facto standard for TCP/IP transport. Yet development in layer 2 protocol sets and manageability extensions to Ethernet standards propose layer 2 packet switching technology as a viable, cheaper alternative to SONET/SDH. Furthermore, native optical transmission is available for higher (1 Gbit/s and up) bandwidth Ethernet standards.

While circuit-switched technology is essentially connection-oriented by design, Ethernet is basically a connectionless protocol. Similar to how TCP/IP can be transformed into a connection-oriented network through the use of MPLS[1] (adding a shim header to TCP/IP packets), connection-oriented functionality can be added to Ethernet by adding a forwarding label to the Ethernet frame. The ongoing design of MPLS TP[2] (transport profile) is a result of a joint effort between ITU-T and IETF. MPLS TP removes features from MPLS not needed for pure connection-oriented operation.

This paper looks at layer 2 switching technology –specifically Ethernet– as a viable alternative to SONET/SDH. Carrier Ethernet[3] offers functionality (such as QoS) typically only associated with circuit-switched carrier-grade transport. Where SONET/SDH offers deterministic multiplexing behavior because of fixed time slotting, frame-based Ethernet uses statistical multiplexing, which influences QoS performance; this is detailed in Section 2. The impact of more complex topology construction typically seen in Ethernet segments (compared to SONET/SDH networks which are often ring-based) is explained in Section 3.

2. INFLUENCE OF STATISTICAL MULTIPLEXING

In this section, we look at the influence of statistical multiplexing on QoS performance. More specifically, we are interested how the presence of other traffic streams in the same forwarding path impacts packet delay jitter. Delay between packet injection and arrival is caused by link propagation time as well as queuing time, the latter being non-deterministic in the case of statistical multiplexing. Because of this, flow delay time is the average of the flow's packet delay times; the packet delay time shows some variance (jitter).

In order to determine the delay jitter on a traffic stream, packet injection and arrival times are logged and correlated. Since off-node time stamping to a separate logging host itself would incur additional delay jitter, timestamping of packet injection and arrival is done on the source and destination nodes respectively.

Real-time clocks on distinct nodes cannot be synchronized to a sufficient level to perform jitter analysis (which requires sub-ms accuracy and precision). As delay jitter is a variance of delay, absolute synchronization between host clocks is not required, synchronization of clock ticking rates however is. In order to achieve this, all hosts synchronize with a NTP server every few seconds. This way, clock adjustments based on NTP timestamps results in kernel software clocks that run at the same speed, compensating for hardware clock drift. This compensation works as long as this hardware clock drift is constant; for example, if a node has been brought up recently, rising internal temperature will influence clock drifting rates, which will be noticeable on sub-ms scale even for the short experiment time runs used in these results (< 1 min.). However, NTP synchronization, temperature control and proper processing of timestamps allows to use off-the-shelf test bed hardware for timing results, instead of more expensive hardware based traffic analyzers.

The nodes and link connectivity are configured on a Emulab[4] based emulation test bed. Nodes run Linux, links are implemented using per-experiment configured Ethernet VLANs. Click Modular Router[5] is used on the nodes for layer 2 switching functionality; extensions to Click development within the Celtic TIGER project[6] support label-based forwarding of Ethernet frames.

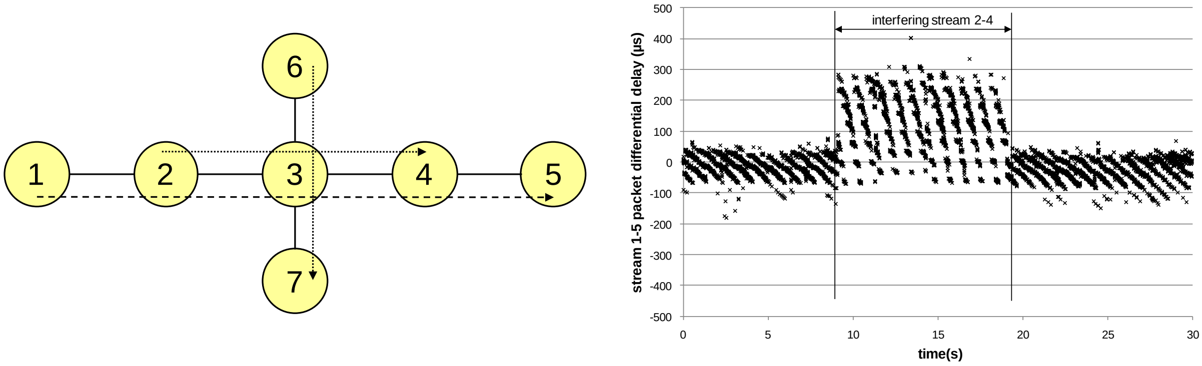

Figure 1. Influence of statistical multiplexing on delay jitter.

In the results on Fig. 1, jitter analysis is shown for a RTP traffic stream, carrying a PAL video stream encoded using H.264 at roughly 2.2 Mbit/s. Since PAL video runs at 25 video frames per second, an RTP burst is sent every 40 ms. Due to the nature of H.264 encoding, the burst size varies dramatically depending on the type of H.264 frame that is being sent (I or P-frame). The RTP traffic stream is sent from node 1 towards node 5 (on the figure). To show the impact of statistical multiplexing, an interfering traffic stream is injected into the topology. This interfering stream is a flow either from node 6 to 7 (crossing traffic flow), or from node 2 to 4 (parallel traffic flow). The interfering stream has a (IP payload) bandwidth of 800 Mbit/s, while the practical maximum capacity of a link is about 940 Mbit/s (as measured, with 1Gbit/s Ethernet line rate). Therefore, the links are run close to their practical maximum, but no packet loss occurs.

For the crossing flow, no influence was noticed during experimentation (not shown on the figure). Nodes have sufficient processing capability to process the frames. Any delay jitter should therefore be caused by queuing effects, as seen on the figure (right) for the parallel flow case. Jitter results for roughly 30 s of traffic are shown, with the interfering flow active for 10 s. The low (relative to line rate) bandwidth video stream shows around 200 µs of delay variance, increasing to about 400 µs when the interfering flow is active. End-to-end flow delay (average of frame delay samples) is higher in this case.

Statistical multiplexing impact seems quite limited; the 400 µs delay variance being a lot smaller than the 40 ms video frame interval. However, this is for simple aggregation (two streams), and for a constant-bandwidth interfering stream. For realistic scenarios, topologies will have to aggregate a large number of video streams which will all be bursty, increasing QoS degradation effects. Originating several streams at the same node will cause these streams to be queued before injection into the network, meaning the number of streams per node as such does not affect jitter caused by multiplexing at intermediate nodes. Only at aggregation points does the statistical multiplexing incur additional jitter. Using the nodes 1 to 5 of the topology on Fig. 1, four nodes can be used as sending nodes, so looking at results, four concurrent streams allow for general conclusions on multiplexing influence (for this topology). The four sending nodes plus one terminating node example will also be used for different topologies in the next section.

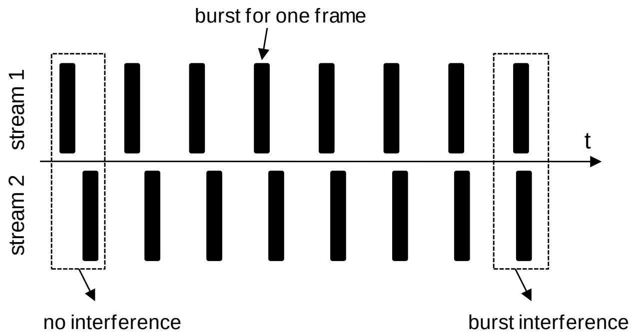

Figure 2. Scaling of video frame burst times to vary extent of inter-stream burst interference.

As the video traffic stream takes only 2.2 Mbit/s, while line rates are 1 Gbit/s, the experiment uses video traffic streams with scaled up burst sizes, in order to increase total bandwidth going over the network: traffic bandwidth / link capacity ratio becomes higher and the statistical multiplexing effect becomes more critical.

Additionally, the burst send times of different streams are scaled (temporally) by a factor close to 1. This results in temporal interference (moiré effect) of the bursts (Fig. 2), which causes fluctuations in used link bandwidth when seen on a very short timescale (i.e., duration of one Ethernet frame or one burst), even if bandwidth averaged over a longer time (several video frames) does not vary. The scaling factor is chosen such that roughly two peaks show up in the burst correlation interference pattern, for the typical duration of the emulation experiments (which is ~30 s). Note that on an emulation platform, it is not possible to synchronize burst sending times in between nodes on the desired time scale (μs); burst send time scaling however allows evaluating all possible synchronization possibilities.

On Fig. 3 then, the effect of statistical multiplexing is shown for 2 or 4 (top/bottom) and ×10 or ×100 bandwidths (left/right). As expected from the temporal scaling of burst send times of one stream versus another, two peaks in the packet arrival delay are seen for the ~30 s experiment. For the 4 stream case, one stream is injected at node 1, 2, 3, 4 each; for the case with 2 streams, sending nodes are node 1 and 3. All streams are terminated at node 5. Furthermore, for the 4 stream case, temporal scaling factors for stream 2 and 3 are assigned uniformly in between scaling factor for stream 1 and 4. Please note the different delay time scales for the 10× vs. 100× bandwidth charts (maximum of 3 and 30 ms respectively).

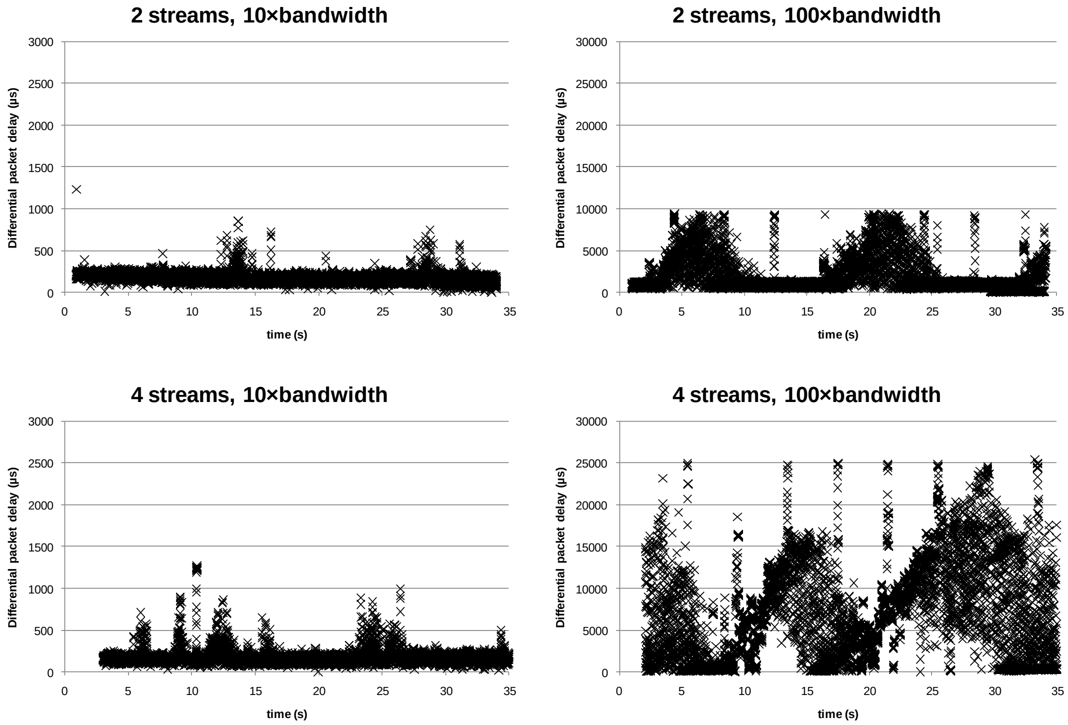

Figure 3. Impact of multiple bursty streams and stream bandwidth

As expected, more streams cause slightly higher delay variance (since more capacity is used); more importantly, the four stream scenario shows 'second-order' peaks in the delay charts. For the ×10 bandwidth case, jitter is minimal, in the order of 1 ms. For the ×100 bandwidth case however, the results show jitter peaks up to 25 ms.

3. INFLUENCE OF TOPOLOGY

In this section, the influence of topology is examined. For video surveillance applications, the output of several video cameras is typically aggregated and terminated at a monitoring station. On Fig.4, the three different types of aggregation topologies are shown. The bus topology is the same as used in Section 2 (Fig. 1), statistical multiplexing happens every time a stream is injected into the network (at each of nodes 2, 3 and 4). It should be noted that, in terms of forwarding, bus topologies are equivalent to ring topologies (as used in e.g. SONET/SDH). For the tree topology, aggregation and multiplexing is done at two levels. The hub case finally aggregates all streams at the same time.

On Fig. 5, again packet delay is shown for the three topologies (left). Also, packet loss vs. bandwidth scaling factor is shown (right). At first sight, one might conclude the hub topology to yield better results in terms of jitter performance. However, it turns out that for this topology the lower jitter is mostly caused by higher packet loss.

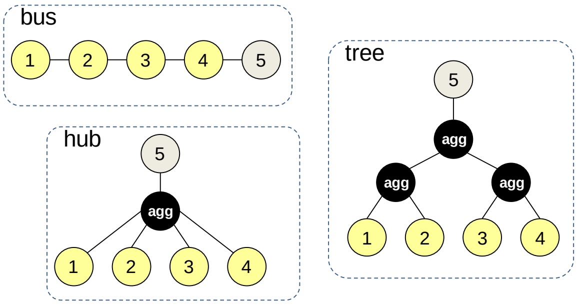

Figure 4. Bus, tree and hub aggregation topologies.

As Fig. 5 suggests, packet loss is reduced as the number of multiplexing steps increases. This is because with each multiplexing step, a new queuing buffer is available. A topology with more aggregation points has a total larger multiplexing buffer memory, so fewer packets overflow this buffer memory and get dropped. Total traffic that is terminated to the destination node approaches the practical capacity of the 1 Gbit/s links (4 streams × 2.2 Mbit/s × scaling factor). The bus topology only starts experiencing packet loss when links are near full capacity, for the other topologies this starts much earlier. For hub, this even starts at a point where nearly two-thirds of line rate capacity is unused.

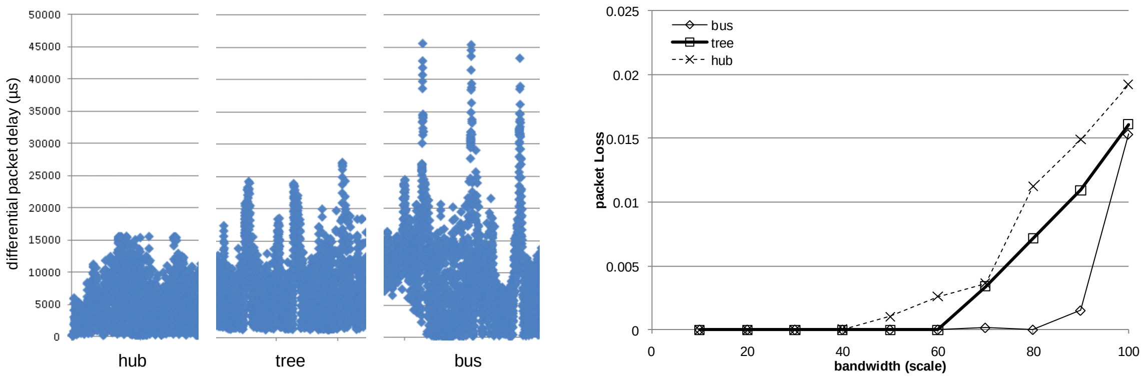

Figure 5. Jitter and packet loss influence of topology.

For the specific case of video stream aggregation, jitter and packet loss are related closely. Due to the repetitive nature of traffic bursts (the entire network 'beats' at the rhythm of video frame burst arrivals, i.e. 25 Hz for PAL video), jitter exceeding the video frame duration (40 ms) threshold will generally be accompanied with packet loss in some part of the network. If the network capacity is considered provisioned as 'statistical' timeslots (repeating every 40 ms), a burst delayed by more than the video frame duration will basically occupy capacity reserved for the burst from the next video frame (in the next 40 ms time interval); in other words, not enough capacity was available in that 40 ms time interval: the network is overloaded.

4. CONCLUSIONS

In this paper, some issues were identified in carrying QoS sensitive services over packet oriented layer 2 networks. We looked at aggregation of video streams over an Ethernet transport. Delay jitter and packet loss were compared for bus, tree and hub aggregation topologies. Bus topologies experience lower packet loss due to larger overall multiplexing buffer capacity.

REFERENCES

[1] RFC 3031, Multiprotocol Label Switching Architecture

[2] RFC 5654, MPLS-TP Requirements

[3] A. Reid, P. Willis, I. Hawkins, and C. Bilton, "Carrier Ethernet", Communications Magazine, IEEE, Vol. 46, Nr. 9, pp. 96-103, September 2008

[4] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb, and A. Joglekar, "An integrated experimental environment for distributed systems and networks", in proc. 5th Symp. on Operating Systems Design and Impl., pp. 255–270 (2002)

[5] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek, "The Click Modular Router", ACM Trans. on Computer Systems, Vol. 18, Nr. 3, pp. 263–297 (2000)

[6] W. Tavernier, D. Papadimitriou, D. Colle, M. Pickavet, P. Demeester, "Emulation of GMPLS-controlled Ethernet Label Switching", in proc. TridentCom, Washington, United States, April 2009