Challenges & Solutions of Mixed Data Rate Ethernet Networks

Presenter: Manfred Kunz, Chief System Architect
Automotive Ethernet Solutions, NXP Semiconductor

Author: Don Pannell, Fellow
Automotive Ethernet Networking, NXP Semiconductor

IEEE Ethernet & IP Tech Days – November 2021
OVERVIEW

• The Challenges
• Historical Solutions
• Today’s Solutions
• Summary
Overview – The Challenges of Mixed Data Rate Networks

- IEEE 802.3 has standardized Automotive PHYs in link speeds from 10 Mb/s to 10 Gb/s
- This variety gives a network designer the ability to use the most cost-effective / power-efficient solution at each point in the network
- A classic client-server architecture model is shown
  - Bridges are used to connect multiple slower links to a single faster link as flows move down toward the server
  - This allows all nodes to talk to the server at the same time without concern for the bridges dropping frames
  - The server can also talk to all the end stations assuming it can keep up with its processing – but faster link to slower link flows create congestion points where bridges can drop packets
  - This model will be used to show the methods available to mitigate dropped packets due to congestion
The Problem – Congestion Points

- Congestion points occur wherever the rate of data entering a port’s Tx Queue’s buffer exceeds the port’s transmit rate.
- This rate change is easy to see if P9 (1 Gb/s) is sending data to P1 or P2 or P3 (all 100 Mb/s) – a 10 to 1 ratio.
- But P1 could also be a congestion point if both P2 & P3 are sending data to it at the same time – a 2 to 1 ratio.
  - This is true for any port where 2 or more ports of the same speed are sending data to a port at the same time.
- Tx Queue buffers are like shock absorbers designed to absorb the impact of momentary congestion.
  - But if the congestion is too long, the buffers will fill, and packets will be dropped! They are like water funnels that can overflow.
- Note: 10BASE-T1S ports are congestion points even if the data entering a 10BASE-T1S port’s buffer is 10 Mb/s.
  - This is because the 10 Mb/s media is shared between all nodes.

Symbols:
- Blue = Tx Queues/Buffer
- Green = MACs
- Orange = Switch Fabric
- Solid line = 100 Mb/s Point-to-Point
- Dashed line = 1 Gb/s Point-to-Point
Historical Solutions

The non- Time Sensitive Networking solutions
TCP/IP’s Built-in per-flow Slow Start Rate Mechanism

- When the Server sends a flow to nodes B, K & P each flow needs to be at a different Tx rate due to link speeds used and due to other network traffic flows
- TCP/IP sends a small burst of packets to each node
  - If the node acknowledges a good reception (ACK), TCP/IP increases the burst size for that flow until the node replies that it didn’t get all the data (NAK - negative acknowledge)
  - TCP/IP periodically re-tests the limits to see if the congestion went away; resulting in additional packet loss
- Benefits:
  - The Server can send data to many nodes back-to-back utilizing its link’s bandwidth as well as dynamically adjusting to the network’s link utilization changes
- Problems:
  - It is slow to stabilize the rate, it is non-deterministic, and it requires packets to be dropped to adjust!
    - This works for TCP/IP as it supports re-transmission of lost data

End Stations, = Bridges
- Color is Speed

- = 10 Mb/s Multi-Drop
- = 100 Mb/s Point-to-Point
- = 1 Gb/s Point-to-Point
- = 10 Gb/s Point-to-Point
IEEE 802.3x MAC Flow Control

- MAC Flow Control uses Pause frames
  - The receiving node uses these frames to tell the sending node to stop transmitting due to its buffers filling up, and when the buffers recover, to re-start transmitting again
  - It is limited to Full-Duplex links only
- Benefits:
  - No packets are dropped if the buffers are large enough to support the round-trip time of sending the Pause (after the Tx line frees up) to the time the flow stops
- Problems:
  - Since a Paused port stops sending all frames, the network can slow down to the speed of the slowest link
    - For example: If the 1st Bridge connected to node B sees its buffer filling it will Pause the 2nd Bridge; which causes the 2nd Bridge’s buffer to fill which causes the Server to be Paused — slowing the Server’s 10 Gb/s link to < 10 Mb/s — for all flows! This happens in the real-world when Bridges are configured to never drop any frames

---

**Diagram Notes:**
- End Stations, Color is Speed
- Bridges
- = 10 Mb/s Multi-Drop
- = 100 Mb/s Point-to-Point
- = 1 Gb/s Point-to-Point
- = 10 Gb/s Point-to-Point

**Network Speeds:**
- 10BASE-T1S
- 10 Mb/s Multi-Drop
- 100 Mb/s Point-to-Point
- 1 Gb/s Point-to-Point
- 10 Gb/s Point-to-Point
IEEE 802.1Qbb MAC Priority-based Flow Control

- Priority-base Flow Control was developed for Data Center Bridging when it was evident its Congestion Notification Protocol (802.1Qau) could not prevent packet drops under quick changes in network congestion
  - It is limited to Full-Duplex links only
  - It was designed to work with Congestion Notification

- Benefits:
  - It is an enhancement to Pause frames where only designated Traffic Class Queues are paused allowing others Traffic Classes to transmit

- Problems:
  - All flows in a given Traffic Class Queue are Paused
    - Which generally contain multiple flows!
  - Hard to configure - there are at most 8 Traffic Classes
Summary of non-TSN Mechanisms

- **TCP/IP Slow Start**
  - Great for the Internet – doesn’t work with UDP, etc.

- **IEEE 802.3x MAC Pause**
  - May be OK “inside a box” (for single flow applications) –
  But “outside the box” it breaks TCP/IP’s Slow Start mechanism & can choke link bandwidth

- **IEEE 802.1Qbb Priority-based Flow Control**
  - Designed for Data Center Bridging where short delays on the data are better than TCP/IPs re-transmission

- **TCP/IP gets each flow to its intended destination at its optimal rate – but can this be done for non-TCP/IP flow types without dropping any packets?**
Today’s Solutions

The Time Sensitive Networking solutions
TCP/IP is a very Interesting Historical Model

- TCP/IP scales as each End Station is responsible to adjust the needed buffering and transmission rate of each of its independent flows
  - It works in the largest of networks today (Corporations & the Internet) – but for TCP/IP flows only
  - Each End Station is responsible for the number of flows it generates as a self-contained system
    - Giving designers a simpler problem to solve, & to verify, as its all local to the End Station Talker only!
  - And it works with simple Bridges for 10 flows or 10,000 flows through the device
    - Bridges don’t need to be per-flow aware (except for policing at the End Station to Bridge connection)
- TCP/IP gets each flow to its intended destination at its (almost) best possible rate

- But TCP/IP has drawbacks for deterministic real-time applications
  - A TSN solution needs to work without dropping any packets as part of the flow rate mechanism
  - And it needs to work with UDP in addition to OSI Layer 2 flows, etc.

- Can we keep the good parts of TCP/IP and improve on its problem areas?
IEEE 802.1Qav – TSN’s Credit Based Shaper (CBS)

• The problem of “sending many streams of data through a network such that no packets are dropped” was solved by the 1st TSN Profile, the plug-&-play AVB use case
  - This solution needed:
    ▪ To support non-TCP/IP flows (e.g., UDP & OSI Layer 2) where re-transmission is too slow
  - And it was accomplished:
    ▪ By reducing the stress (on the buffers) at all network congestion points for a given class of flows

• CBS ensures no packets are dropped due to congestion by adding a specific requirement for end station Talkers:
  - Clause 5.20 b) of IEEE 802.1Q-2018: “Support the operation of the credit-based shaper algorithm (8.6.8.2) as the transmission selection algorithm used for frames transmitted for each stream associated with the SR class.”

  - This additional requirement for Talkers over Bridges, performs per-flow shaping (to limit each flow’s transmission rate independently like TCP/IP does) followed by a per-class shaper (to de-burst each class’s data – a common requirement for both Bridges & Talkers)

Note: CBS does not require network-wide time awareness, meaning it works without gPTP (IEEE 802.1AS)
IEEE 802.1Qav – A Talker’s Credit Based Shaper (CBS)

- The figure below shows an example of the CBS requirements in a Talker
  - All Classes (A, B, C & D) have the same structure – i.e., per flow rate-limiting by CBS, merging into a Traffic Class queue where the aggregate is de-burst by the Traffic Class’s CBS
  - Only Class A’s 4 Flow Queues are shown, but each Class can have any number as needed
- Since Automotive is outside the AVB Profile (IEEE 802.1BA) Automotive can define more than AVB’s 2 Classes (A & B) & it can define their Observation Intervals to be different

- This is a model only – there are many ways to implement this
  - Buffers for Flow 1 to 4 are needed, but Class A’s buffer can be virtual
  - When frames appears on the wire is what is important!
  - Software implementations can be used for mid to low-rate flows
Why is per-flow Rate Shaping Required in a Talker and not a Bridge?

- Consider the Server being a Talker sending:
  - Class A Flow 1 to End Station B at 2 Mb/s
  - Class A Flow 2 to End Station K at 20 Mb/s
  - Class A Flow 3 to End Station P at 200 Mb/s
  - A total of 222 Mb/s for Class A
  - The Server’s Class A CBS setting = 222 Mb/s

- Also consider without per-flow CBS, if the Server:
  - Builds a large burst of frames for End Station B
  - And places these frames into Class A’s queue ahead of frames intended for Flow 2 and/or Flow 3
    - Likely, as CPU’s work on one task at a time for a period
  - Bridge 1 will receive Flow 1’s burst at 222 Mb/s and will transmit that burst at < 10 Mb/s to End Station B
  - Depending on the burst size, frames will be dropped!
Comparing Per-Flow CBS vs. Per-Class CBS in a Talker

- In a single process time slot, the CPU links in a burst of 2 Mb/s packets for End Station B
  - Talker model the frames go in Flow 1’s queue
  - Flow 1’s CBS releases them at 2 Mb/s
  - Bridge model the frames go in Class A’s queue
  - Class A releases them at 222 Mb/s!

- In a subsequent process time slot, the CPU links in a burst of 200 Mb/s packets for End Station P
  - Talker model the frames go in Flow 3’s queue
  - Flow 3’s CBS releases them at 200 Mb/s
  - Bridge model the frames go in Class A’s queue
  - Class A releases them at 222 Mb/s!

- Flow 1 Tx = 2 Mb/s, Flow 3 Tx = 200 Mb/s
  - Flow 1 Tx = 222 Mb/s, Flow 3 Tx = 222 Mb/s
Common Questions about Talker CBS

• Why not add more buffering to Bridges (Bridge 1 specifically in the example)?
  - This doesn’t scale: The memory size that works for one application probably won’t for another
    ▪ The buffer size may work for 2 to 1 congestion, but then fails with 3 to 1 or higher congestion
  - Not possible: Memory can’t be added to self-contained single chip Bridges

• What is the purpose of the Class CBS in a Talker?
  - If Flow 1, 2 & 3 are all 30 Mb/s, Class A is 90 Mb/s. Class A’s CBS helps spread out the 3-frame burst if a frame from each Flow showed up in Class A for transmission at, or near, the same time

• How does this solve congestion?
  - If the Talker transmit each flow at its intended rate, congestion point are de-stressed
    ▪ The only remaining contentions are a single lower priority interfering frame, and frames at the same priority. But Bridge buffers can handle these small-size contentions

• Can IEEE 802.1Qcr, Asynchronous Traffic Shaping (ATS), solve this?
  - ATS in Bridges is great. But ATS rate limits flows so large bursts from a Talker will still blow out a Bridge’s buffers & frames are dropped. Talkers still need to Tx each flow at its expected rate!
Summary & Conclusions
Summary & Conclusion

• Talker per-flow rate transmission control solves the problem
  - And the Credit Based Shaper (CBS) is the only TSN Tool defined for Talker per-flow rate control
• CBS was standardized in 2009 so it is mature & available in many products
  - CBS does not require network-wide time awareness, meaning it works without gPTP (802.1AS)
• Per-flow CBS does not always require hardware as it is a frames/second problem
• Some flows are self-shaping, and these flows don’t need the a per-flow CBS queue
  - For example, audio flows from a microphone collects $n$ samples and then transmits a frame, collects $n$ more samples and then transmits the next frame, … at a constant frames/sec rate
• Do bridges need CBS in small networks?
  - CBS allows small bursts of packets to “catch up” due to momentary contention
  - The per-class CBS function in Bridges, de-burst these small bursts so they don’t get larger
  - In my opinion, due to the small size of Automotive networks, per-flow Talker CBS is all that is required if Bridge hops are few, as the Talker is the critical place to get the flow rate correct!
• Solution: Per-flow CBS in Talkers + CBS in Bridges (for larger networks)