IEEE P802.1DC and IEEE P802.1Qdv Time Sensitive Networking Developments

Norman Finn
Huawei Technologies Co. Ltd
nfinn@nfinnconsulting.com
finn-tutorial-802-1DC-1Qdv-2023-12-v02.pdf

This presentation represents only the opinions of the author. It has not been approved by IEEE 802.1.

Two works in progress in IEEE 802.1 TSN TG

IEEE P802.1DC Quality of Service Provision by Network Systems

- In Standards Association ballot, the last ballot/comment cycle.
- Allows a system that is ***not*** a bridge to claim conformance to all of the IEEE 802.1 TSN quality of service standards, along with YANG modules to support them.

IEEE P802.1Qdv Enhancements to Cyclic Queuing and Forwarding

- In Task Group ballot, the first ballot/comment cycle.
- Provides multi-level nested transmission cycles, with output bins selected according to time-of-arrival, per-flow byte count, or (potentially) a field in the frame.

IEEE P802.1DC Quality of Service Provision by Network Systems

IEEE 802.1Q TSN Feature list (for bridges)

Queuing/dequeuing methods

- Strict priority. Devil take the hindmost.
- Enhanced Transmission Selection (e.g. weighted fair queuing) ensures low-priority some access.
- CBS: Per-priority Credit-Based traffic Shaping.
- Scheduled output. Queues enabled/disabled by a repeating schedule.
- Cyclic Queuing and Forwarding. Whole network advances TSN traffic one hop at the same time.
- Asynchronous Traffic Shaping. Each frame assigned an earliest output time, based on arrival time, using per-flow and per-group state machines.
- Potentially, the enhanced CQF described in IEEE P802.1Qdv.

Related mechanisms

- General Flow Control and Metering. Token bucket red/yellow/green marking.
- VLAN tagging, including 8-level priority and 2-level (green/yellow) drop eligibility.
- Priority Flow Control. Better than XON/XOFF per-priority control to prevent input buffer overflow.
- Frame Replication and Elimination for Reliability. Replicates flows on different paths, and later, eliminates redundant packets.
- Per Stream Filtering and Policing. Time scheduled per-flow input gates, frame filtering, flow identification for QoS.
- Frame preemption. Interruption/resumption of one frame for more time-critical transmissions.

IEEE 802.1Q TSN Features

More importantly than the provision of any given feature, IEEE Std 802.1Q specifies an overall model that defines the interactions among all of these features.

Any combination of priority, ATS, CQF, round-robin, or other transmission selection can be configured. The resultant behavior of the system is well-defined. YANG modules are provided to control them.

Frame preemption is integrated into the mix; TSN time-critical streams can be preempted by even more critical TSN streams.

IEEE 802.1Q TSN features

This feature set is frequently implemented on devices that are not just IEEE Std 802.1 Q bridges, and often, on devices that have no bridge functionality at all.

None of the QoS features in IEEE Std 802.1Q actually require that the core of Bridge functionality (i.e. forwarding frames based on MAC addresses) be present at all.

The IEEE Std 802.1Q QoS features are rather well isolated from the rest of the document, confined to relatively few clumps of text.

Hence, IEEE 802.1 elected to start work on IEEE P802.1DC, to make these QoS features easily referenceable by standards defining hosts, routers, firewalls, load-sharing appliances, security appliances, top-of-rack switches, NICs, or any other kind of device.

IEEE P802.1DC contents

IEEE P802.1DC D2.0 contains the following:

- A **guidebook** to lead the reader of P802.1DC through the parts of IEEE Std 802.1Q (and its amendments and some related IEEE 802 standards) that are relevant to providing QoS.
- A list of **features**, and a trail of **requirements** for each feature, specifying exactly what parts of IEEE Std 802.1Q (and others) must be satisfied to claim compliance to that feature, even though the feature is not being implemented in an IEEE Std 802.1Q Bridge.
- The **YANG modules** required (most via pointers) for a non-bridge device to control any or all of those features.

IEEE P802.1Qdv Enhancements to Cyclic Queuing and Forwarding

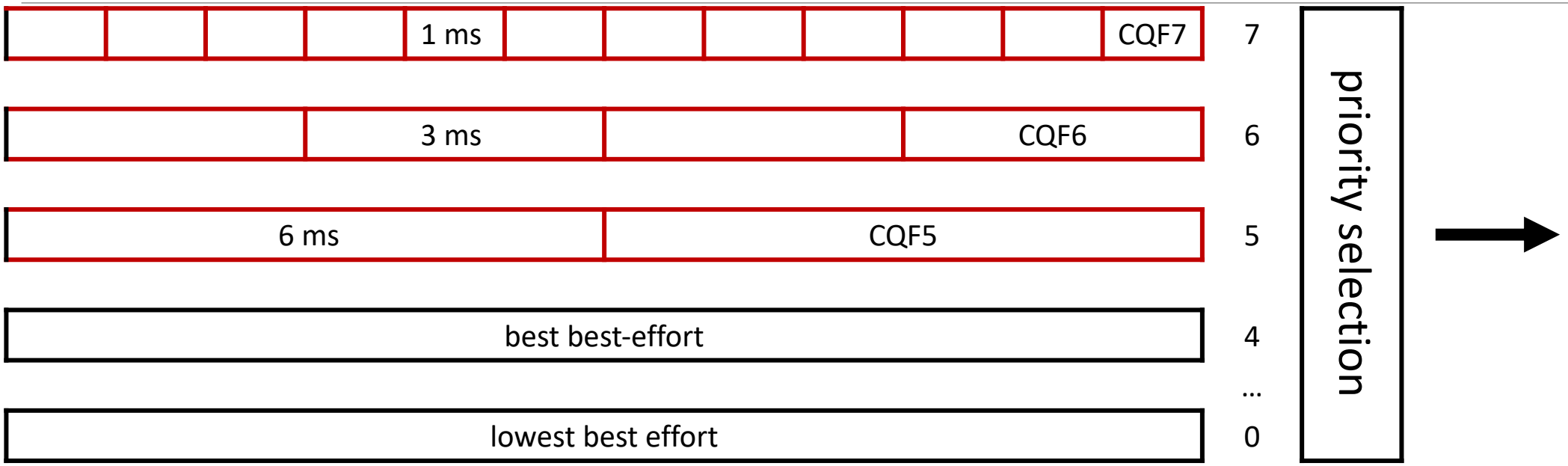
Scaling

- TSN does not scale well to large networks. The reason depends on in part on the queuing method(s) employed. The two most common issues:
  - Adding or removing a single flow can require recomputing the delivery parameters of a large fraction of the existing flows in the network.
  - Adding or removing a new flow can require configuring parameters and/or a state machine in every relay node along its path.
- IEEE P802.1Qdv offers a basis for a variety of related queuing strategies that can reduce these issues.

We will discuss:

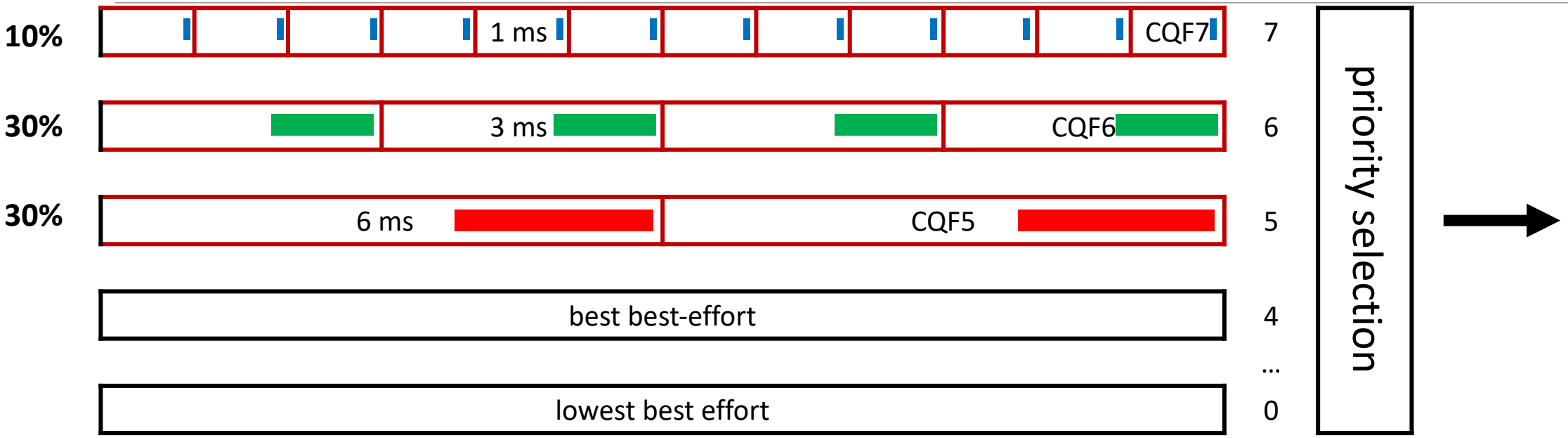
1. The **output** side: Multiple queues, each divided into bins, outputting the bins in rotation, each queue at a different frequency.
   - Bandwidth allocation, buffer allocation, output timing.
   - The timeline paradigm.
   - Why this scales well.
2. The **input** side: Assigning received queues to bins.
   - Three methods: time of arrival, per-flow byte counters, frames marked with a bin number.
   - Scaling tradeoffs: Bin assignment method; time synchronization options.
3. Consequences

Output side
Bins and CQF: timeline view of queues



- Three levels. Fastest cycle time has highest (most important) priority.
- Cycle times are integral multiples, bin swapping is aligned vertically.

Bins and CQF: bandwidth allocation

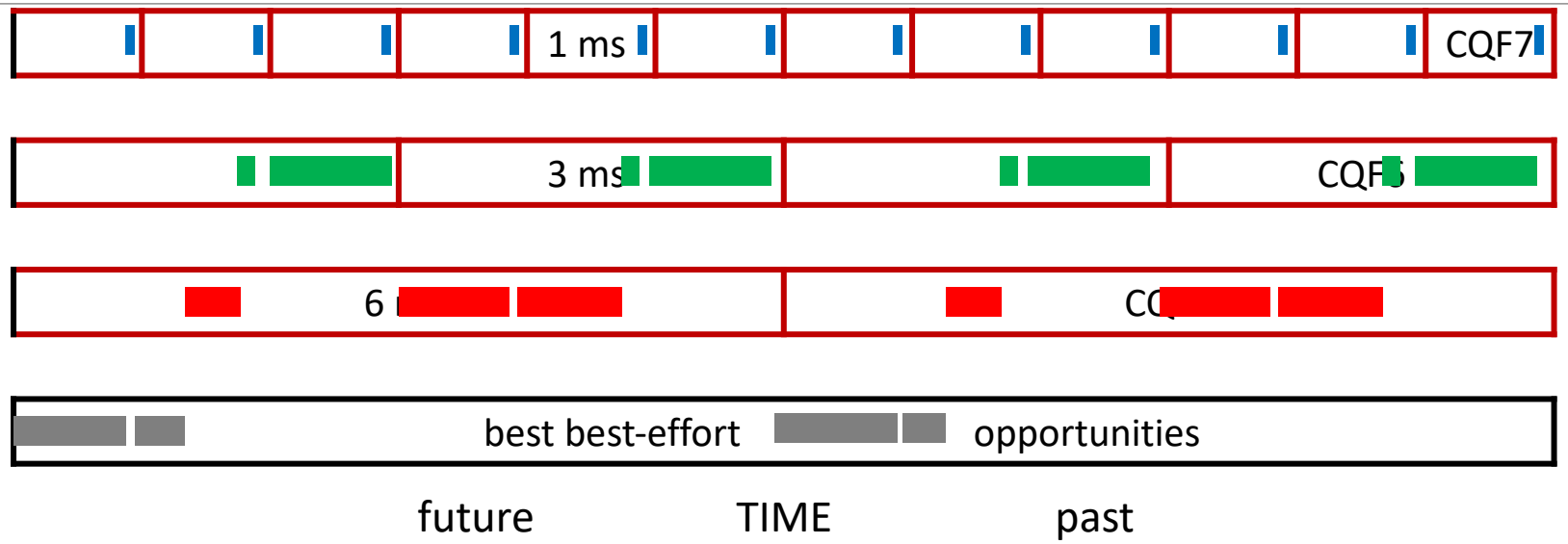


- Bandwidth allocated to a flow as bytes per cycle at some priority.
- Multiple priority levels can be used simultaneously.
- Allocating 40% of priority *p* consumes 40% of total bandwidth.

Bins and CQF: buffer utilization

- Perhaps CQF7 uses extra buffers to allow for input delay variation.
- Perhaps CQF5 uses synchronized clocks to use only 2 buffers.
- Flows can be forced to use use extra buffers to equalize path delays.

Bins and CQF: output timing



- CQF7 is output with highest priority, then CQF6, CQF5, then highest-priority best-effort.
- This makes best-effort lumpy. Additional fine-grained scheduling could smooth out lumps, but at some computation/implementation cost.

Scalable computation load for new flows.

- Admission control is trivial:
  1. For any given port, the total number of bytes allocated over all flows passing through that port, at a given priority, must not exceed (some configured fraction of) the number of bytes that can be transmitted during one bin transmission time.
  2. The total allocated over all priorities cannot exceed the link's (some configured fraction of) the link's bandwidth.
- Per-flow worst-case delay computation is trivial:
  1. Number of buffers used per hop is either fixed or varies by 1, depending on bin assignment method. Time per buffer is fixed to some accuracy.
  2. Physical link delay changes very slowly. (More on this, later.) Per-flow worst-case delay computation is trivial:
- Adding/deleting a flow does not change any other flow's delay.

Isn’t there a downside?

- Flexibility is not infinite (a tradeoff against computation time).
  1. For a flow whose characteristics match a priority’s bin cycle time (bandwidth = one frame per cycle), the efficiency of CQF in terms of bandwidth utilization and minimum delay is optimal.
  2. Supplying multiple cycle times at multiple priority levels allows fitting arbitrary flows to approximately the right priority level.
     a) Too-high priority sacrifices allocable bandwidth, but gives extremely low delivery time. In the extreme, this supports “alarm” flows.
     b) Too-low priority gives worse delivery time, and uses more network buffer space.
  3. For service providers, a fixed repertoire of service choices is not unfamiliar.
- We will get to whether time synchronization is required on a later slide.

The timeline paradigm

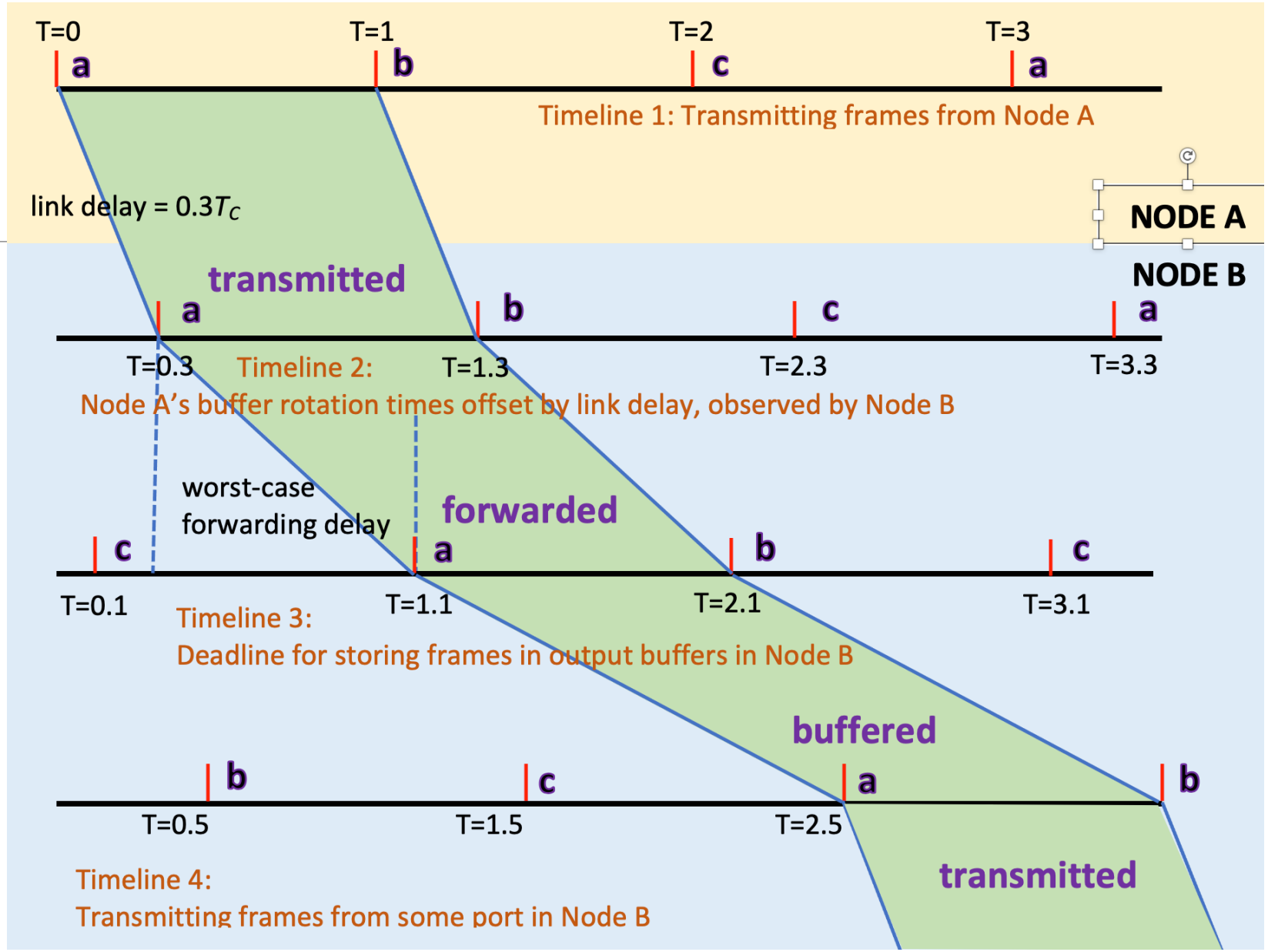
- What is critical is that, at each hop, all of the CQF flows are arranged on a transmission timeline that adheres to the CQF rules:
  - Bins rotate to the transmitting state regularly.
  - Bins at different priorities on the same port are integrally aligned.
  - No bin is over-allocated, so every bin empties before its time expires.
- This ensures that interference among all priority levels is bounded and easily calculated.

Input side
Selecting an output bin [0..*n*]

(We are looking at one priority level. Bin rotation frequencies are at least approximately the same at every hop. Phase of rotation can vary along a path and among ports on one relay node.) **Three methods:**

1. Bin selection based on bin number from previous hop
   a) Obtained by time-of-arrival of frame (in P802.1Qdv draft 0.4)
   b) Obtained by a field in the frame (*not* in P802.1Qdv draft 0.4)
2. Bin selection based on counting bytes stored in output bin so far (in P802.1Qdv draft 0.4)

Timing for bin assignment based either on time of arrival or a field in the frame



Bin selection based on previous hop's bin

If the selection of the next-hop bin is derived from the last-hop bin selection, then **no per-flow state machines, and no per-flow configuration**, is required. Furthermore, the end-to-end **delivery time is constant**, modulo one bin rotation cycle.

But, this requires that all hops along the path rotate their bins at **exactly the same frequency** – that is, the difference in the number of bins output between two hops, over an arbitrarily long period of time, must be bounded.

Bin selection based on previous hop's bin

Getting the network nodes to rotate bins at exactly the same frequency is not trivial, *but it is easier than synchronizing time across a network to high precision.* This is because the bound on the maximal difference in bin count is on the order of **one bin time**, not one nanosecond.

IEEE P802.1Qdv Draft 0.4 includes a simple one-way protocol to align the phase of the transmit and receive clocks (timelines 1 and 2 in the diagram). **This permits the link delay and/or output bin phase alignment to vary slowly.** Of course, this requires buffering in the receiver sufficient to accommodate the variation.

Bin selection based on byte counts

Mick Seaman's Paternoster algorithm, included in P802.1Qdv Draft 0.4, uses a byte counting state machine, per flow, per output, port to ensure that no flow exceeds its allocation. This provisioning affects scaling negatively, but there is no interaction between flows, so no massive recomputation is ever necessary. These counters can be used in several ways:

- With minimal buffering, but with over-provisioning proportional to clock inaccuracies, to provide TSN service without frequency lock.
- At flow ingress or at frequency lock regional boundaries to recondition flows.
- To support changes in bin rotation frequency (e.g. when link speed changes).
- To support flow aggregation and dis-aggregation.
- As an insurance check within frequency-locked regions.

Mixed bin selection

Each relay node along a flow's path can use a different number of bins per output queue, and a different bin selection method.

One output bin can be fed by input ports and/or flows using any combination of time-based, label-based, or count-based output bin selection methods.

One (multicast) flow can be sent to different output ports using different bin selection methods.

Flow aggregation

Clearly, flow aggregation is the key to massive scaling in network size. Some relevant points:

- No multi-layer flow wrapper protocol has been proposed in IEEE 802.1 for TSN. However, flow identification methods defined by IEEE 802.1 **do** support implicit flow aggregation.
- The bin rotation queue mechanism can be used when aggregating flows, with the aggregation as a virtual output link. This is not necessary, but it can reduce buffer requirements at the dis-aggregation point.
- P802.1Qdv is a good fit for flow aggregation, because it delivers the aggregated flow at a steady rate; this minimizes the buffering required for the dis-aggregated flows at the end of the pipe. (In a frequency-locked region, buffering is optimal.)

Closing remarks

Summary

- IEEE P802.1DC provides a convenient reference for any device to use all of the TSN QoS features.
- IEEE P802.1Qdv is intended to provide a complete specification for “Time Division Multiplexing over packet networks” for small networks, and a foundation for the same for very large networks.

Pointers

IEEE P802.1DC Draft 2.0 “Quality of Service Provision by Network Systems”

IEEE P802.1Qdv Draft 0.4 “Enhancements to Cyclic Queuing and Forwarding”. Annex Y contains the most accessible description of the work.

Seaman, M., “Paternoster policing and scheduling” cr-seaman-paternoster-policing-scheduling-0519-v04.

Finn, N., “Multiple Cyclic Queuing and Forwarding” new-finn-multiple-CQF-0921-v02.

Thank you