IEEE 802.1 Time-Sensitive Networking (TSN)

Norman Finn, IEEE 802.1CB, IEEE 802.1CS Editor
Huawei Technologies Co. Ltd
norman.finn@mail01.huawei.com

Geneva, 27 January, 2018

Before We Start

- This presentation should be considered as the personal view of the presenter, not as a formal position, explanation, or interpretation by IEEE 802.1.

The problem that created TSN

- Time-Sensitive Networking applications are those in which a networked computer system must **respond to external stimuli within a fixed, and often small, period of time**.
- There are a number of widely-used application-specific real-time digital interconnect standards, e.g. **HDMI** (video), **CAN bus** (automotive), **Profibus** (industrial).
- The volume of **802 Ethernet** products and IEEE 802's history of **backwards compatibility** and **regular upgrades** in speed and network size make Ethernet attractive to users of application-specific networks.
- **Problem**: Ethernet is perceived as random in nature, and therefore unsuitable for time-sensitive applications.

TSN’s capabilities

1. Synchronize time to < 1μs accuracy.
2. Ensure zero congestion loss and bounded end-to-end latency to TSN data streams, and bounded interference to non-TSN data streams, via resource reservation.
3. “Hitless 1+1” redundancy against equipment failures.
4. Maintain 100% of the compatibility, scalability, robustness, speed, and reliability that make Ethernet attractive.

1. Time synchronization

- Not all TSN applications require precision time synchronization.
- For those that do need it, standards based on IEEE Std 1588 are recommended.
- TSN supplies one such standard, IEEE Std 802.1AS.
- Other 1588 profiles are compatible with TSN.

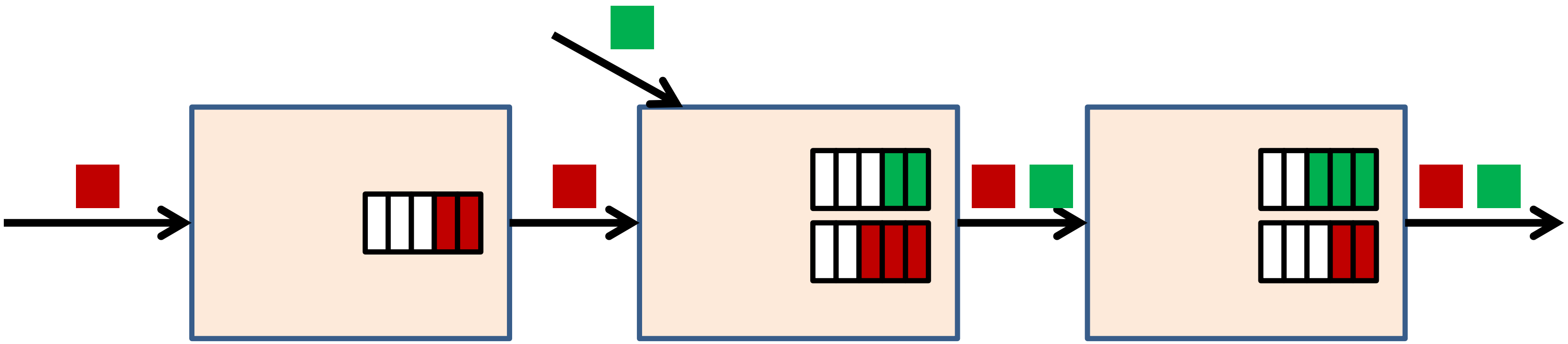
2. Zero congestion loss, bounded latency

- Zero congestion loss + finite number of buffers in the network = bounded latency.
- Neither can be provided to an unlimited load.
- A ”reservation”, a contract between the transmitter and the network, is required:
  - The transmitter specifies a not-to-exceed bandwidth (max packet size, measurement interval, max number of packets) and latency requirements.
  - The network either reserves the resources necessary to meet these requirements, or refuses the reservation.
  - **Average latency is unimportant. Worst-case end-to-end latency is critical.**

Achieving bounded latency

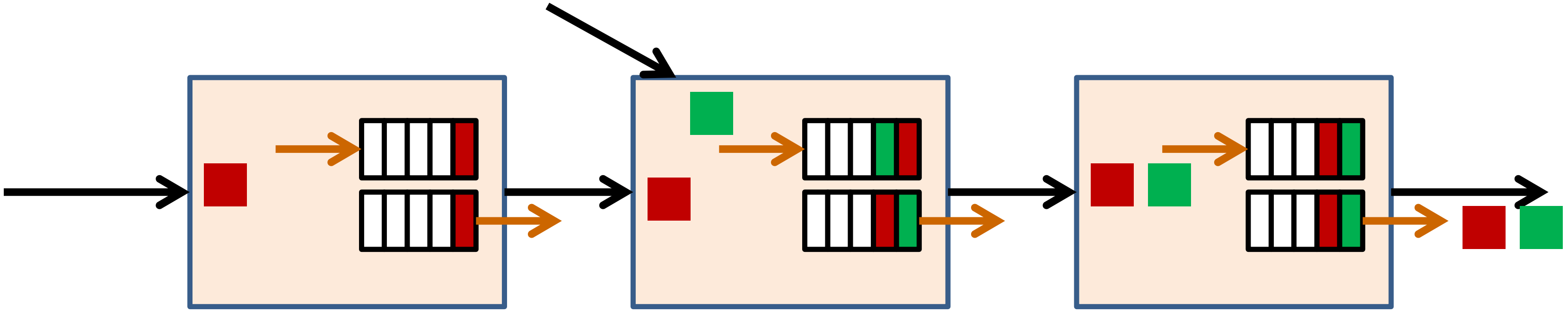
- Methods standardized and in progress:
  - Traffic shaping in output queues.
  - Synchronized multi-buffer swapping.
  - Time-scheduled output queues.
- Congestion feedback methods (e.g. TCP, or throttling the source to avoid congestion) are **not an option** for these applications.

Zero Congestion: traffic shaping



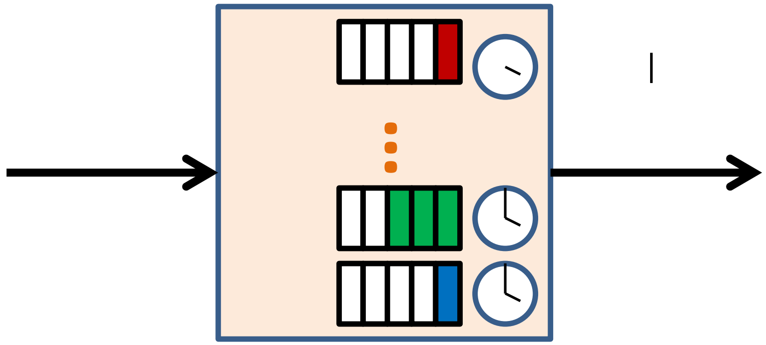
- Each flow can be shaped == output at that flow’s reserved rate.
- It is **not** good to output a packet too soon because the link is idle, because **sending a packet early requires that the next hop have buffer space to hold it**.
- Outputting a packet early is not helpful to TSN applications. Outputting a packet on time means that each hop knows what resources it requires.

Zero Congestion Loss: Cyclic Queuing and Forwarding



- One buffer on each output port receives input from all input ports.
- The other buffer outputs to the link.
- All buffers in all bridges swap roles at the same (time-synchronized) moment.

Zero Congestion and low jitter: Time-scheduled outputs



- A repeating, time-synchronized schedule controls all queues (TSN and non-TSN) on a given port.
- Schedules are defined to 1 ns, but implementations vary in accuracy.
- Any on/off combination is OK. E.g.: all off to drain link, two queues on can compete for priority, all on is the same as no schedule.

Making reservations

- Talkers and listeners may or may not be TSN-aware.
- TSN streams can be unicast or multicast.
- Reservations can be made statically, and changed only when all TSN applications are idle.
- Reservations can be made dynamically, adding or deleting some reservations, while other reservations are carrying data.
- Dynamic reservations can be made using:
  - Decentralized control via peer-to-bridge-to-peer protocols
  - A centralized network controller that can make global optimizations.
  - Application controller(s) either separate from, or integrated with, the network controller.

3. Hitless 1+1 redundancy

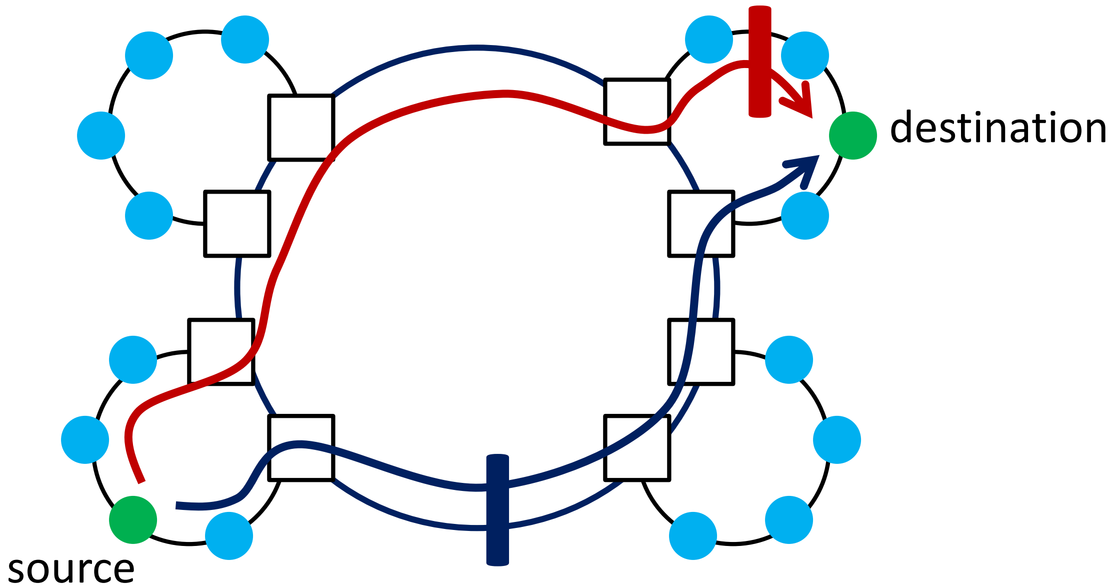
- For most TSN applications, a general-purpose bridging or routing protocol such as IS-IS takes too long to restore connectivity after a failure (or recovery).
- For some TSN applications, a ring protocol or fast-reroute capability provides adequate network availability.
- For other TSN applications, something more is required.

Frame Replication and Elimination for Reliability (FRER)

1. Provide a **sequence number** for every packet in the stream.
2. Replicate each packet, and send the replicated streams over two (or more) **fixed paths** towards the destination(s).
   – Paths may be set up statically or with ISIS, but once set up, they do not shift with link failures.
3. Near the destination(s), keep track of what packets have and have not been seen, and **eliminate the duplicates**.
   – That is, the receiver does not switch between streams. It looks at every packet to ensure that exactly one copy of each is passed on.

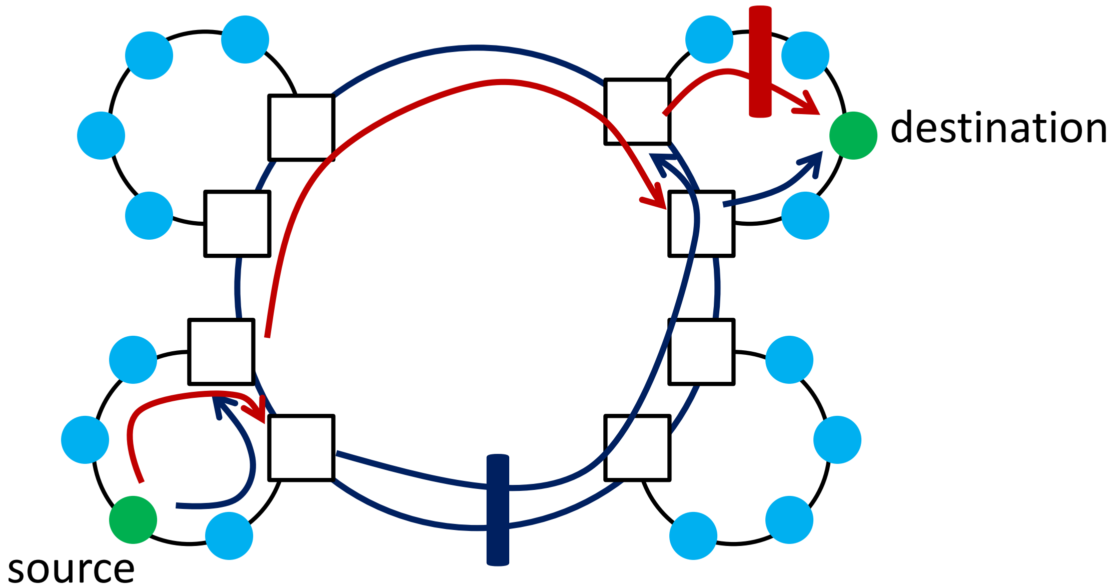
FRER: end-to-end

- It takes two failures to prevent delivery.



FRER: multiple replications

- Multiple failures can often be overcome.



4. It is still Ethernet

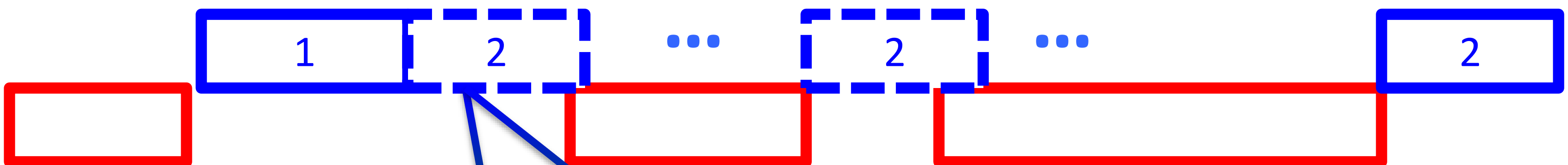
- No special “TSN MAC”. (But, new PHYs to meet new application challenges.)
- TSN and non-TSN traffic can share the same network—non-TSN traffic cannot invalidate TSN guarantees, and TSN can be limited in its effects on non-TSN traffic.
- Robustness against bit errors is maintained.
- Transmission preemption assists sharing.

IEEE 802

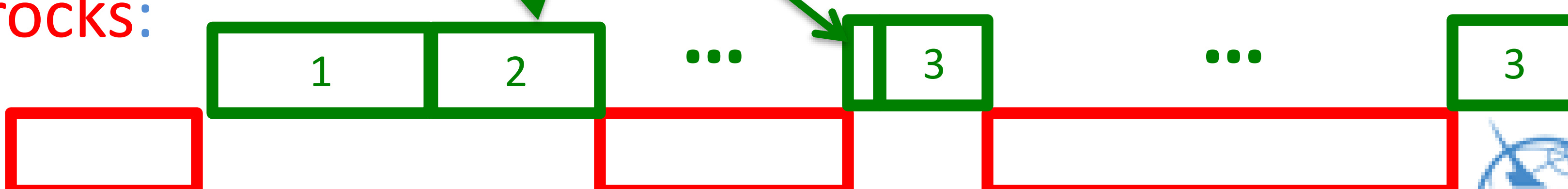
Frame Preemption

- Express frames suspend the transmission of preemptable frames (**802.3br and 802.1Qbu**)
- Scheduled rocks of TSN packets in each cycle:

- Conflict excessively with non-TSN packet rocks:



- Problem solved by preemptive sand between the rocks:



Summary

- Features:
  - Time synchronization
  - Zero congestion loss, bounded latency
  - Hitless 1+1 redundancy
  - Full compatibility with non-TSN applications
- Growing number of applications / markets
- Work underway in IETF to include routers (DetNet Working Group)