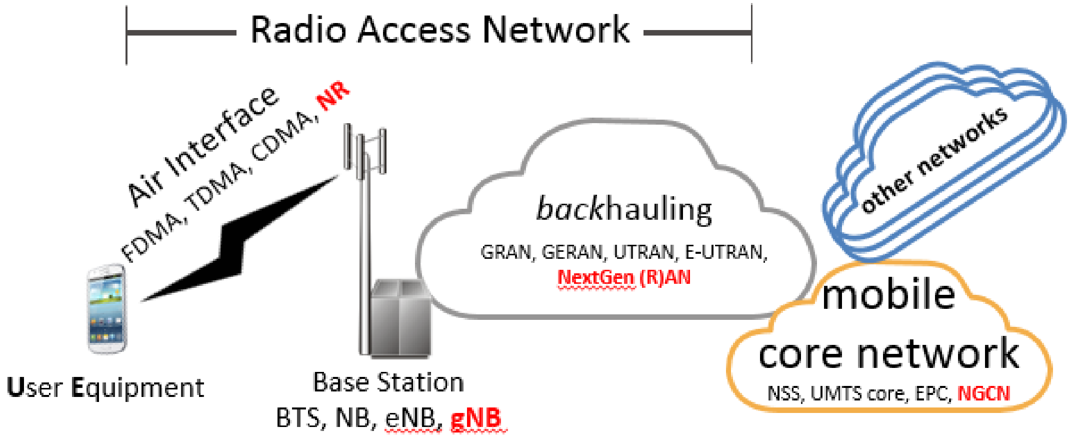
5G xHaul

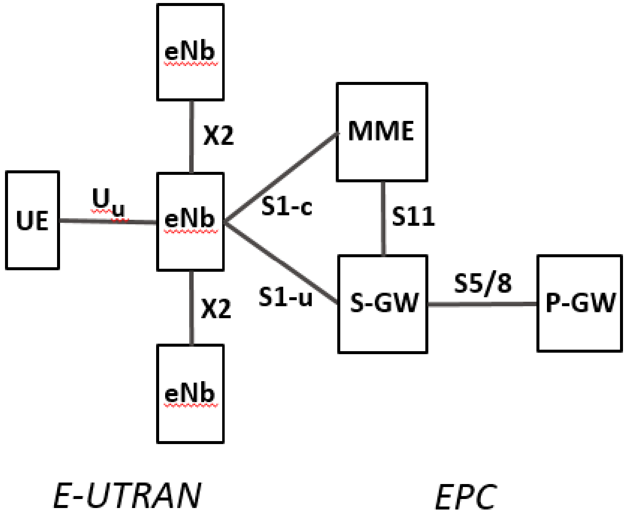

Radio Access Network
Air Interface
FDMA, TDMA, CDMA, NR
backhauling
GRAN, GERAN, UTRAN, E-UTRAN,
NextGen (R)AN
other networks
mobile
core network
NSS, UMTS core, EPC, NGCN
User Equipment
Base Station
BTS, NB, eNB, gNB

Reminder: 4G back/front-haul

We remember that in 4G there were backhaul and fronthaul

Backhaul transports 2 interfaces:

- S1 from the eNB to the core (divided into S1-U and S1-C)
- X2 between eNBs, used for
  - smooth handoff
  - CoMP



X2 data rates are lower (typically <5% of S1)
but delay constraints may be much higher
(< 10 ms, the lower the more efficient)

It is expensive to physically interconnect eNBs
so in practice, X2 is mostly a logical interface
backhauled up to an aggregation point
and then back down to another eNB

4G fronthaul is between

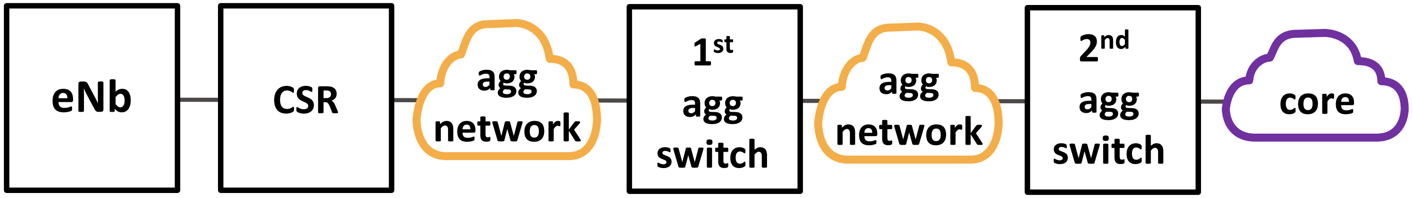
- **R**emote **R**adio **U**nit (AKA **R**emote **R**adio **H**ead)
- **B**ase**B**and **U**nit

4G backhaul in practice

3GPP specifications do not detail *how* to backhaul S1 interfaces

In practice, there is a backhaul network with multiple network elements

- **C**ell **S**ite **R**outer or **C**ell **S**ite **G**ateway
- 1^{st} aggregation switch
- 2^{nd} aggregation switch



The physical layer of the aggregation networks may be

- fiber (including **P**assive **O**ptical **N**etworks) mostly at 1 Gbps rates
- microwave
- DSL

in p2p, tree, ring or mesh topologies

The higher layer may be pure IP, Ethernet, or various flavors of MPLS

In the higher aggregation network OTN and WDM may be deployed

5G xHaul

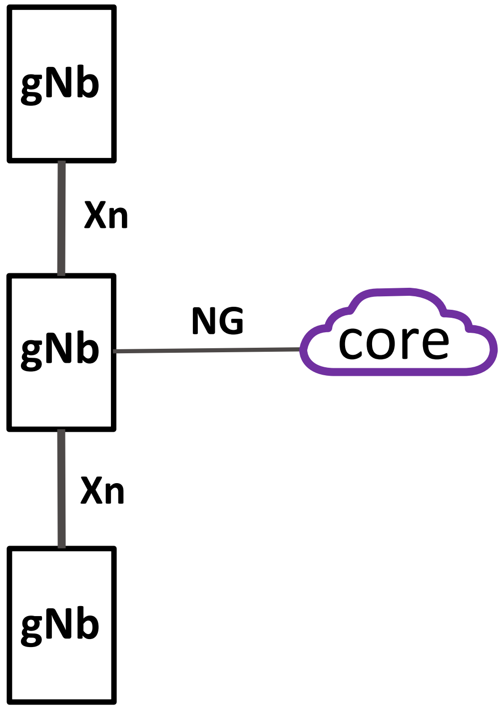
At the highest level of abstraction a 5G network consists of two entities:

- 5G base-station (gNB)
- 5G core network (5GCN)

The interface between the gNBs and the core is called the NG interface

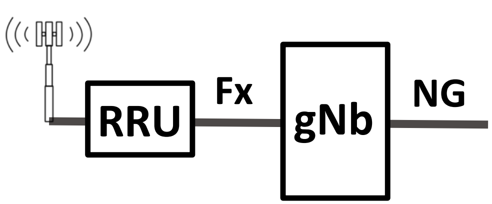
The gNBs can be interconnected by interfaces named Xn

The transport of data over either the Xn or NG interface
is conventionally considered *back*haul



5G also defines *front*haul
between RRU and the rest of the gNB
although this option is being developed by ORAN
and not by 3GPP

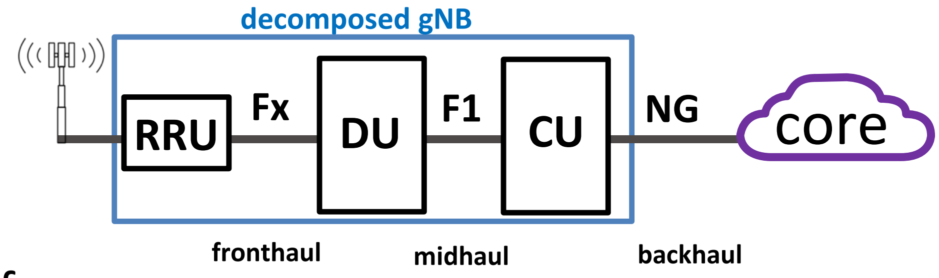
The interface is called Fx



But 5G allows further decomposition of the gNB!

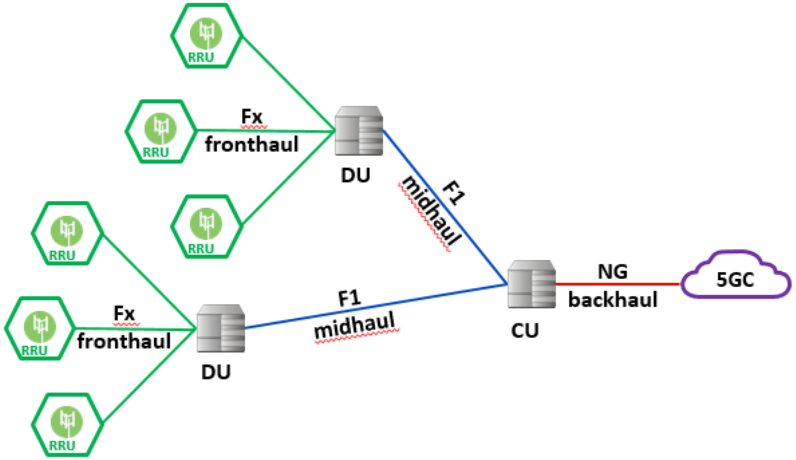
CU and DU, RRU

A 5G gNB *can* be decomposed into
- Central Unit (CU)
- Distributed Unit (DU)
- Remote Radio Unit (RRU)



Thus creating a new F1 *mid*haul interface

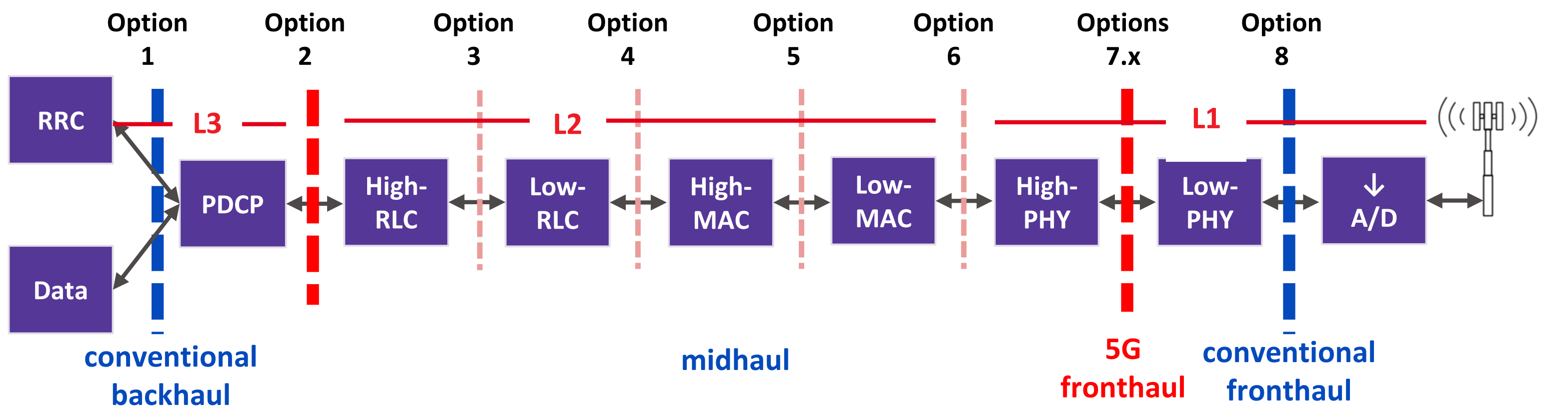
5G allows multiple options, e.g.,
- a unified gNB
- RRU and a combined DU+CU
- a single CU connected to multiple DUs (similar to BBU hosteling)



Functional split options

How do we split the gNB to create RRU, DU, and CU ?

There are various functional split options



3GPP is standardizing split option 2

ORAN is standardizing split option 7.2

- **PHY** – Physical layer processing (L1)
- **MAC** – Medium Access Control (L2)
- **RLC** – Radio Link Control
- **PDCP** – Packet Data Convergence Protocol (L3)
- **RRC** – Radio Resource Control

Each option presents tradeoffs of

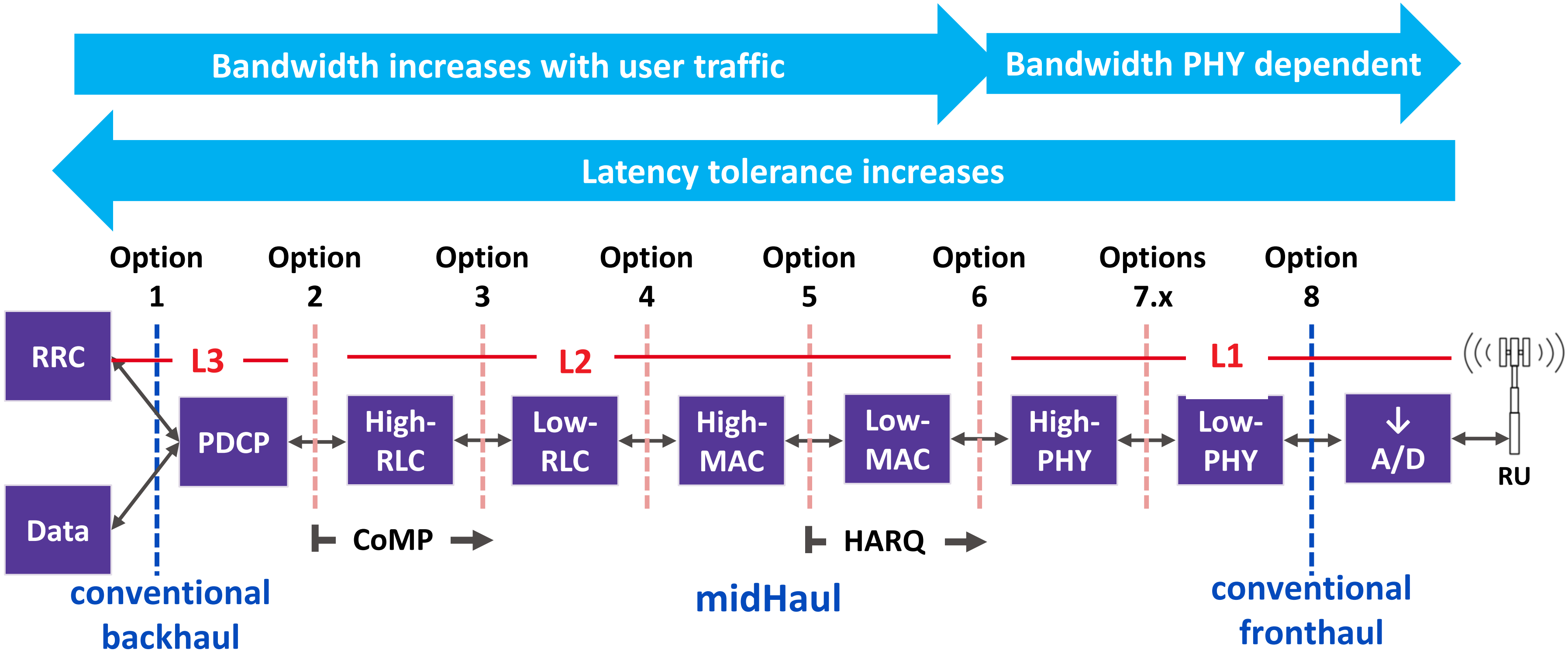
- unit placement – real-estate, computational power, energy requirements
- transport data-rates
- latency
- reliability and resilience
- control/management

Functional option tradeoffs

Higher number (lower level) options

- require higher data-rates (at PHY layer independent of user traffic)
- tolerate less latency

Specific features are located before or after specific split options



Transport challenges

The 3 main challenges are:

- data-rate

  Backhaul estimates are 5 Gbps per cell for initial eMBB deployments

  this will increase over time as users desire higher rates

  Physical layer splits may involve frightening data rates

  The *sampling theorem* tells us that we need to sample at least twice the BW

  so a single **100 MHz** signal requires 200 Msps or 3.2 Gbps (without overhead)

  Assuming 3 sectors each with **16 MIMO** antennas : > **150 Gbps**

  Assuming 3 sectors each with **256 MIMO** antennas : > **2.5 Tbps**

  Assuming **1 GHz** bandwidth, 3 sectors, **256 MIMO** antennas : > **25 Tbps**

  for comparison EoY 2016 the *entire* Internet was 100 PB/month ≈ 300 Tbps

- latency

  URLLC requires low end-end latencies (as low as 1 ms 1-way delay)

  and xhaul above HARQ point always requires low latencies!

- reliability (including very low **P**acket **L**oss **R**atios)

  URLLC requires >5 nines reliability

Why very low PLRs?

BE Internet Packet Loss Ratios can approach 1%
Current cellular air interface induces much worse
Carrier Ethernet can give PLRs of 10^{-6} !

What applications need any better than that?

Example 1 : industrial failures

- assume that 2 consecutive packets lost to a machine in a factory
  may cause the machine to jam, halting production
- GbE may sustain 1.5 Mpps, or about 10^{11} packets per day
  so a factory with 10 operational networks handles 10^{12} packets per day
- a PLR of 10^{-6} under IID means a 2-packet loss ratio of 10^{-12}
  so there will be a production halt every day!

Example 2 : audio-video mastering

- mastering an 8K (7680*4320) at 60 fps requires 3.8GBps = 2.5 Mpps
- a camera to storage PLR of 10^{-6} this means 2.5 packet losses per second!
- since latency rules out TCP, recording is impossible

Potential 5G RAN transport technologies

Recent transport innovations can assist supporting these requirements

- 10GbE, XGS-PON
  - but 10 Gbps is only satisfactory for initial 5G deployments
- 25 GbE (802.3by), 1-lane 50 GbE (802.3cd)
  100/200/400 GbE (802.3bs, 802.3ck)
- FlexE
- **M**obile (**M**ulti-access) **E**dge **C**omputing
- Synchronization (SyncE, IEEE 1588, DGM)
- Network slicing
- **T**ime **S**ensitive **N**etworking (and **Det**erministic **Net**working)
- **F**rame **R**eplication and **E**limination (IEEE 802.1CB)

high data rates

low latency

ultra high reliability

Time/Frequency Synchronization

Frequency and time accuracy requirements are defined
to assure efficient and proper functioning of the air interface

RAN timing requirements are becoming stricter from generation to generation

Base stations obtain timing from the RAN
(unless they have a local source of timing, e.g., GNSS)

5G requirements will be at least as strict as 4G
and some experts are speaking of them becoming significantly stricter

So the 5G RAN must deliver ever more accurate timing!

We will shortly see that new xhaul requirements
may be solved assuming accurate time at every network element (TSN)

How do we deliver sync over an asynchronous packet network ?

There are 2 important technologies

- SyncE – frequency synchronization of Ethernet physical layer
- 1588 (**P**acket **T**ime **P**rotocol) – time of day distribution over Ethernet or IP
  - IEEE 802 AVB WG has produced an AVB profile called 802.1AS
  - ITU-T SG15/Q13 has produced telco profiles G.8265.1, G.8275.1/2

Higher rate physical layers

Ethernet physical layer rates are typically multiples of 10
- 10Mbps, 100Mbps, 1Gbps, 10Gbps, 100Gbps, ...

The present 100Gbps standard is based on 4 *lanes** of 25 Gbps
- so it is natural to allow the use of a single lane
  - for rates higher than 10 Gbps

Single lane 25G has been a standard Ethernet rate since 2016

For yet more flexibility, the FlexE standard enables m*25G

IEEE is working on increasing the lane speed to 50G and eventually to 100G
- resulting in 4 lanes with capacity of 200G and eventually 400G
- which FlexE can further bond together

* depending on standard, a lane may be a pair/fiber or may be a λ

Time Sensitive Networking

TSN and **Det**erministic **Net**working are Ethernet and IP/MPLS technologies that enable:

- *very low* and *guaranteed* packet propagation *latency*
  - time aware scheduling/queuing
  - time coordinated forwarding
  - frame preemption
- *very high reliability*
  - zero congestion loss (PLR of 10^{-10})
  - resource reservation
  - seamless redundancy
- dynamicity – flows can be removed or added w/o impacting other flows
- co-existence of TSN traffic with regular traffic
  - up to 75% express traffic

for applications such as IIoT, V2x, fronthaul

TSN is being developed for Ethernet by a task group in IEEE 802.1

DetNet is being developed for IP and MPLS by the DetNet WG in the IETF

Some TSN Components (some already in 802.1Q-2018)

Latency
- 802.1Qav (clause 34) Forwarding and Queuing Enhancements
- 802.1Qbu, 802.3br Frame preemption
- 802.1Qbv Scheduled traffic
- 802.1Qch (Annex Q) Cyclic queuing and forwarding
- 802.1Qcr Asynchronous shaping

Reliability
- 802.1CB (standalone) Frame Replication and Elimination
- 802.1Qca Path control and Reservation
- 802.1Qci Per-stream filtering and policing
- 802.1AX-rev LAG revision

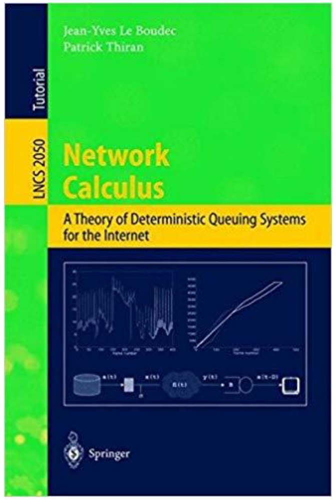
we won't discuss *all* of these

Resource Management
- 802.1Qat (clauses 34, 35) Stream reservation protocol (SRP)
- 802.1Qcc SRP enhancements and performance improvements
- 802.1Qcw Yang data models for Qbv, Qbu, and Qci
- 802.1CS Link local reservation protocol

Misc
- 802.1AS Timing (including 1588 profile)
- 802.1CM TSN for mobile fronthaul
- IEEE 1722 Layer 2 Transport Protocol for TSN

Time Sensitive Networking

TSN and DetNet ask how we can improve network performance
if we have highly accurate synchronization (say, better than 1 μsec)
at network elements (Ethernet switches, IP/MPLS routers) ?



We'll see that we can

- significantly *reduce* latency
- achieve *bounded* latency

for time sensitive flows

Much of the TSN/DetNet work is based on the *Network Calculus*
a mathematical theory dealing with deterministic queuing
such as in communications networks

TSN and DetNet support co-existence of sensitive and non-sensitive traffic
(sensitive traffic can be up to 75% of the total load)

TSN uses a control protocol (SRP) for configuring switch time behavior

802.1Qbu Frame preemption

The major source of residence latency for a high priority packet
is its waiting in a queue for a packet being output to complete transmission

For example, assuming a 1500 B packet just started transmission
the high priority packet needs to wait:

- 10 Mbps 1.7 msec
- 100 Mbps : 170 μsec
- 1 Gbps : 17 μsec
- 10 Gbps 1.7 μsec
- 100 Gbps 0.17 μsec

and the situation will be much worse with 9K jumbo packets
which are being standardized

It is possible with present protocols to *runt* the outgoing packet
but this would require its retransmission
and burden switches along the path to parse and discard it

A solution to this problem (not the most important element in TSN!)
involves *preemption, frame fragmentation,* and *local reassembly*

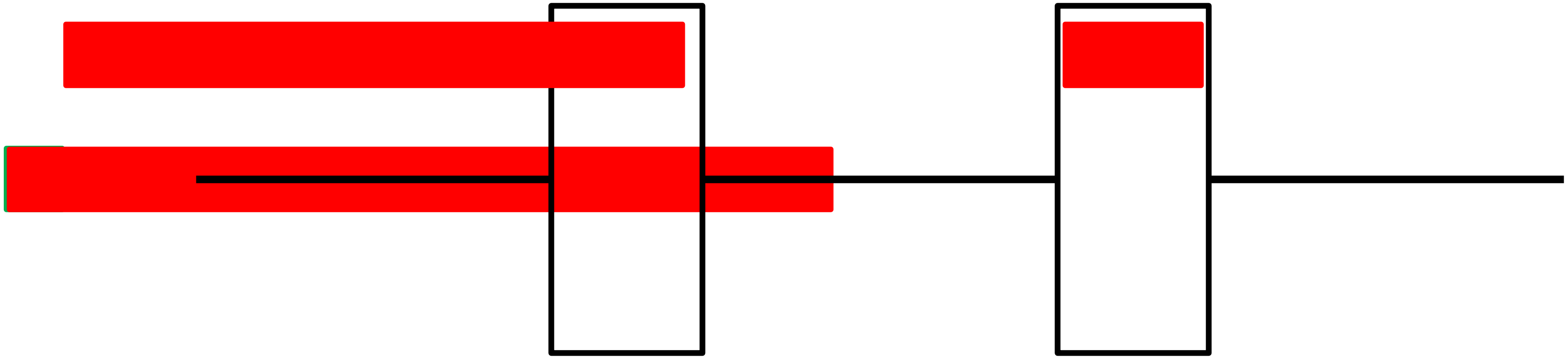
802.1Qbu Frame preemption (cont.)

Published in 2016 and being absorbed into 802.1Q

When express frame(s) arrives and a normal frame is being transmitted

- the packet transmission of the normal frame is temporarily suspended
- the *neighboring* switch buffers the content received
- the express frame(s) are sent and forwarded
- the transmission of the normal frame is continued
- the neighboring switch reassembles the outgoing packet and forwards

Optimal bandwidth utilization of background traffic for time aware shaping and low-latency communication in non-scheduled networks



802.3br Interspersing Express Traffic (IET)

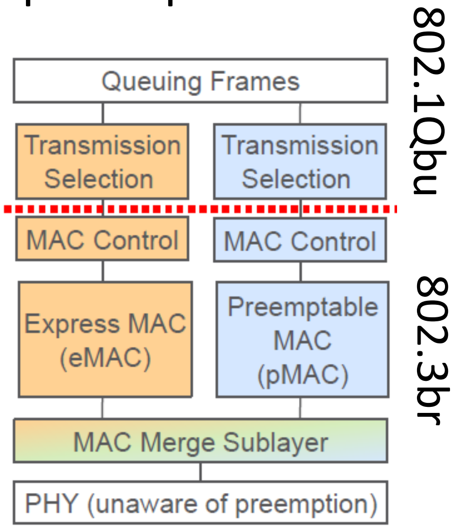
802.3br provides Ethernet physical layer support for frame preemption

All frames are classified as either *express* or *preemptable*

Frame could be preempted multiple times
    but no nested preemption

IET support is discovered via LLDP (new TLV)

Express frames have the usual preamble
  but a special physical start-of-frame
  SMD-e = 55 instead of SFD = AB



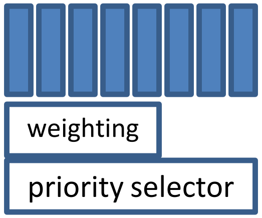
Frame fragments

- are at least 64 bytes (multiple of 8 except last fragment)
- have usual preamble but special start-of-frame
  - first fragment has SMD-Ix = 66 CC FF 33 (for frame 0 1 2 3)
  - non-initial fragments have SDM-Cx = E1 D2 1E 2D + fragment ctr
- have a 0..3 fragment counter
- have their own (modified) FCS
- are not valid MAC frames for non-compliant devices

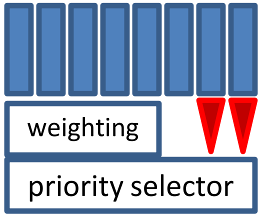
802.1 Queuing

802.1Q-1988 defined *strict priority* selection

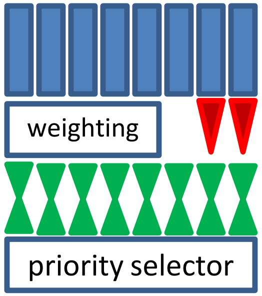
802.1Qaz (absorbed into 802.1Q-2012)
added *weighted queues*



802.1Qat adds *credit-based* fair queuing
(borrowed from RPR)
transmit frames when non-negative *credit*
credit increases when frames wait



802.1Qbv adds *time-sensitive* queues
queues are Open or Closed
according to *timeslots*



802.1Qat Credit Based Queuing

802.1Qat adds credit-based fair queuing (borrowed from RPR)

- a Credit Based Shaper (CBS) spaces out frames to reduce bursting
- a frame is only transmitted when its queue has non-negative credit
- credit
  - increases at rate idleSlope when frames are waiting
    or no frames waiting but credit is negative
  - decreases at rate sendSlope when frames are transmitting
  - rates are adjustable (per queue results in weighted queuing behavior)
- shaped queues have highest priority (higher than unshaped queues)
- Qat still guarantees bandwidth to the highest unshaped priority
- CBS is similar to burst rate shaper
  but with useful mathematical properties
  - only parameter is bandwidth
  - impact on queue of adjacent shapers
    = impact of 1 shaper with total BW

802.1Qbv Scheduled Traffic

Published in 2015 and being absorbed into 802.1Q

Qbv introduces Time Aware Traffic Shaping (Time Sensitive Queues)

- requires that every network element
  has highly accurate time (e.g., from 1588)
- time-gated egress CoS queues transmit one at a time
  based on a precise *timeslot* schedule
- implemented by circular collection of *time aware* gates

Directly timing release of packets can
support *scheduled* applications (e.g., process/vehicle control)
provide latency and PDV guarantees
completely avoid congestion
return to TDM-like determinism

Qbv retains credit-based shapers for *non-scheduled* applications

Time Sensitive Queues

In Qbv **all** CoS queues (not just the TSN queues) are cyclically gated
so non-TSN traffic is also released in timeslots

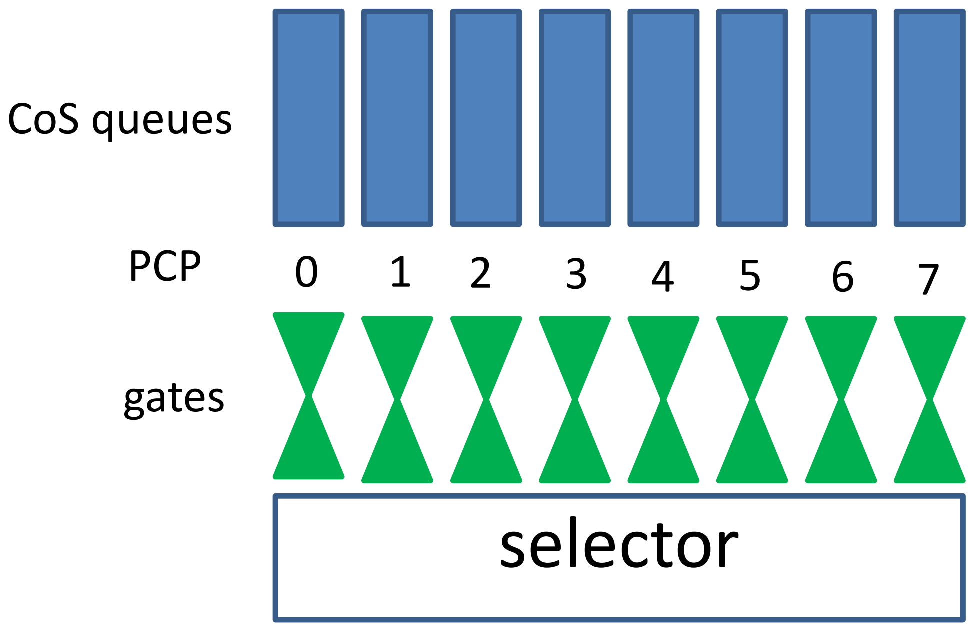
Qbv generally results in bursts of packets belonging to a stream

Timeslot schedules are dynamically computed
by a centralized management system
that configures the network nodes using SRP

Schedules are specified with up to 1 ns granularity
(although implementations may be less precise)
thus PDV can be reduced to about 1 ns

Schedules might require guard times between TSN timeslots
unless preemption is used

Qbv Scheduler



Gate Schedule

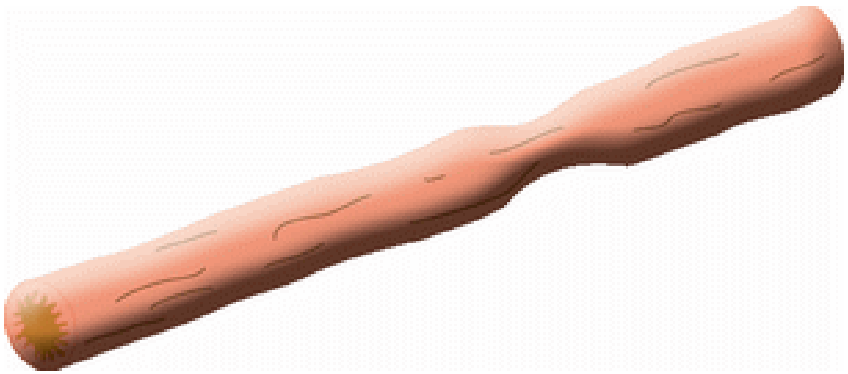
T	0	1	2	3	4	5	6	7
0	O	C	O	O	C	O	O	O
1	C	O	O	O	O	O	C	C
2	O	C	C	C	C	O	O	O
3	O	O	O	O	C	C	C	O
N	*repeat*							

Peristaltic Queuing

What can we do with Qbv time-gated queues?

Knowing in detail each scheduled flow's behavior
we could configure every NE in the entire network
to be free precisely when the next scheduled packet arrives
(this can be done using 802.1Qav)

But there is a simpler (but less optimal) method
called peristaltic queuing



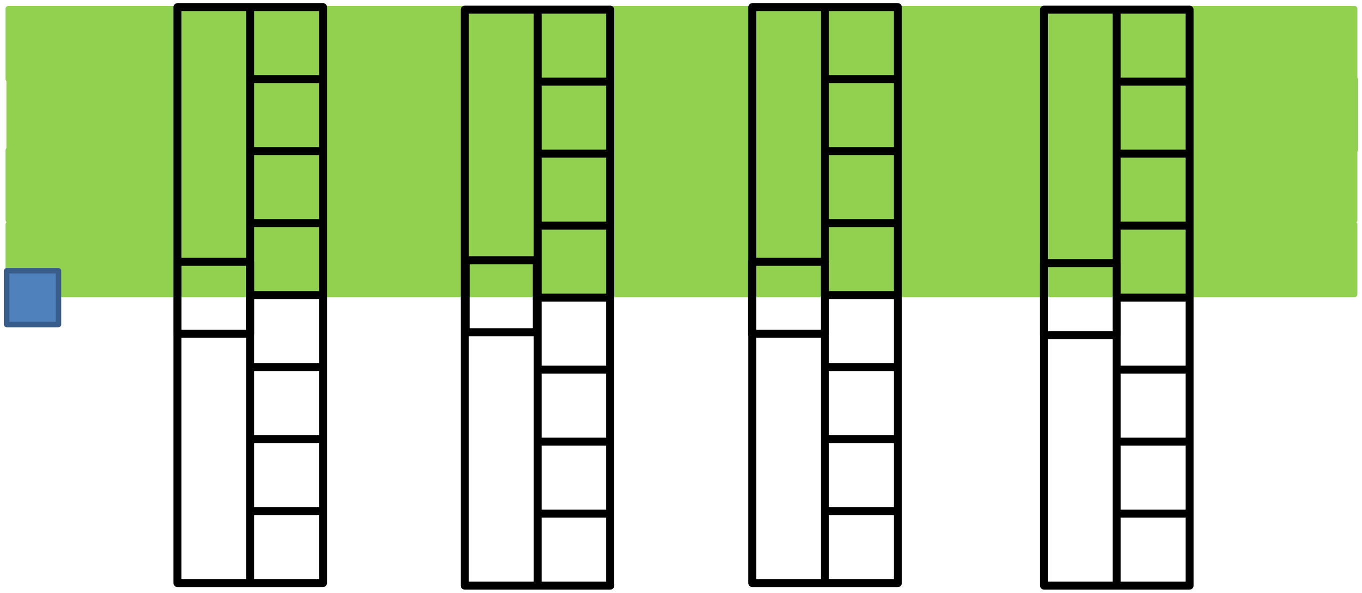
802.1Qci/Qch Cycling Queuing

802.1Q-2018 Annex Q (CQF, formerly called peristaltic shaping)

Exploits accurate timing to provide without intricate signaling
defined (*nonoptimal*) upper latency bound

Each switch collects frames according to PCP
and forwards all packets of the same traffic class in *one cycle*

Maximum bytes allowed per clock cycle configured by SRP



Stream Reservation Protocol

SRP reserves network resources for flows (similar to RSVP-TE) and uses a Traffic Specification (Tspec)

802.1Qat published in 2010, absorbed into 802.1Q-2011

SRP is used to configure all of the TSN features

SRP specifies the required resources and enables their dynamic maintenance and facilitates registration, deregistration, and retention of resource reservation information in relevant network elements

- *Talker Declarations* are distributed along the path(s) to *Listeners*
- *Listener Declarations* run back and actually reserve resources

802.1Qcc proposes enhancements to SRP as defined by Qat

- support for 1000s of streams
- configurable SR (stream reservation) classes and streams
- reduce chattiness and CPU load for handling SRP
- supports L2, but probably won't be used for L3 (DetNet)

802.1Qav FQTSS

Forwarding and Queuing Enhancements for Time-Sensitive Streams

Approved by AVB TG in 2009 and is now clause 35 of 802.1Q-2014

- planning and admission control (CAC) based on SRP
- procedures for mapping priorities to traffic classes
  - class A - latency ≤ 2 msec
  - class B - latency ≤ 50 msec
  - control traffic (AS and SRP)
  - best effort traffic
- precise synchronization based on 802.1AS
- CBS shaping of streams at source based on 802.1Qat
- identification of non-participating devices

802.1CB Frame Replication and Elimination

Published in 2017 **Frame Replication and Elimination for Reliability**

Basically a packet-by-packet 1+1 protection mechanism

- no failure detection needed
- insert stream ID and sequence number into every FRER frame
- frames are replicated at source
- duplicates are eliminated at destination

Several stream identifier methods:

- DA + VID
- SA + VID
- DA + VID + IP SA + IP DA + DSCP + ...

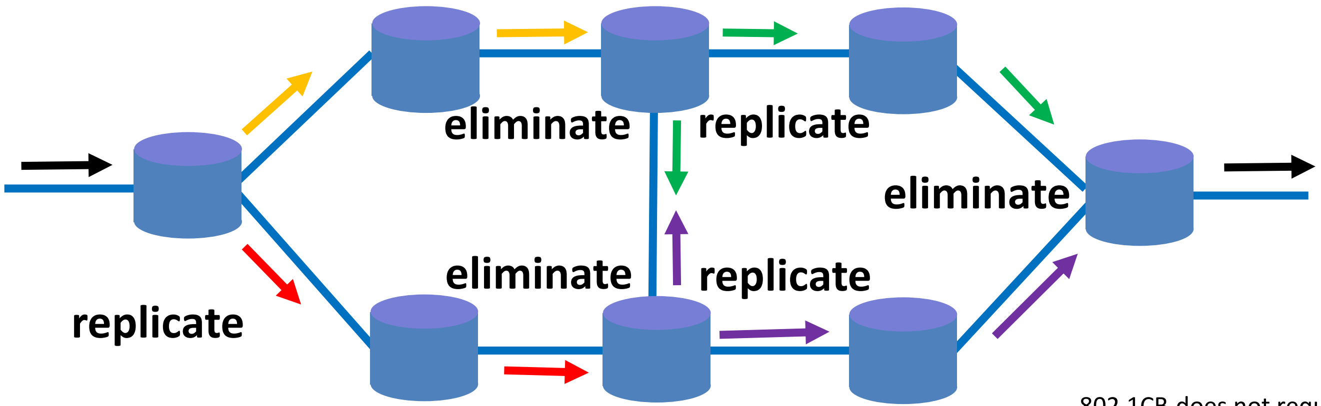
Several sequence number methods:

- new L2 sequence number tag for Ethernet frames
- HSR or PRP sequence numbers
- Ethernet PW sequence numbers

Re-replication and Re-elimination

But replicating at source and eliminating at destination
only protects against a single network failure
(except source and destination points themselves
which are out of scope)

FRER allows configuring replication and elimination
at intermediate nodes



802.1CB does not require in-order delivery
because it assumes large buffers
and packets may be re-ordered

802.1CM TSN for Fronthaul

Published in 2018

Joint work with the *CPRI Cooperation*

Specifies requirements for transporting time-sensitive fronthaul streams over Ethernet networks
    including delay and sync (with calculations)

Specified for CPRI (*class 1*) [encapsulation not specified]
    and eCPRI (*class 2*)

CPRI flows (IQ data, C&M control/management, sync)
    are handled as separate flows

Synchronization via SyncE and ITU-T telecom profile (BC or TC)

Adopts eCPRI's *categories* (A+, A, B, C) and specifies 2 profiles

802.1CM Profiles

Profile A

- use legacy Ethernet equipment
- strict priority Ethernet bridge
  - IQ data with high priority traffic class
  - C&M data with lower priority
- use 2000 bytes for all frames
- MEF 10.3 shaping

Profile B

- assume *very minimal* TSN functionality is present
- Qbu+3br preemption enabled bridge
  - IQ data is express traffic
  - C&M data is preemptable
- 2000 bytes frames for IQ frames, flexible for others
- MEF 10.3 shaping

Mobile Edge Computing

Mobile (**M**ultiaccess) **E**dge **C**omputing offers another solution
to both data-rate and latency requirements

MEC enables terminating traffic close to the gNB or first aggregation nodes
rather than backhauling all the way to the core

By providing processing power close to the UE
network congestion and latency are both reduced

Some MEC applications

- Internet breakout
- DNS caching
- Content Delivery Networks
- mobile big data analytics
- fog networking (IoT processing)
- connected car (V2x)

MEC concepts have been absorbed into 5G's **S**ervice **B**ased **A**rchitecture

Why slicing in 5G?

5G can support coexistence of traffic with wildly diverging requirements by using *network slicing* in the air interface, xhaul, and core

- *on-demand* assignment of networking/computational resources
  - bandwidth, forwarding tables, processing capacity, etc.
- resources can be physical or virtual, but
- each slices acts as a strongly isolated network or cloud
  - isolation of management, security, and performance

3GPP defines a slice by

- Slice/Service type (SST) expected slice behaviour (features and services)
- Slice Differentiator (SD) optional information to differentiate between slices of the same Slice/Service type

Slice/Service type	SST value	Characteristics.
eMBB	1	slice suitable for the handling of 5G enhanced Mobile Broadband.
URLLC	2	slice suitable for the handling of ultra- reliable low latency communications.
MIoT	3	slice suitable for the handling of massive IoT.