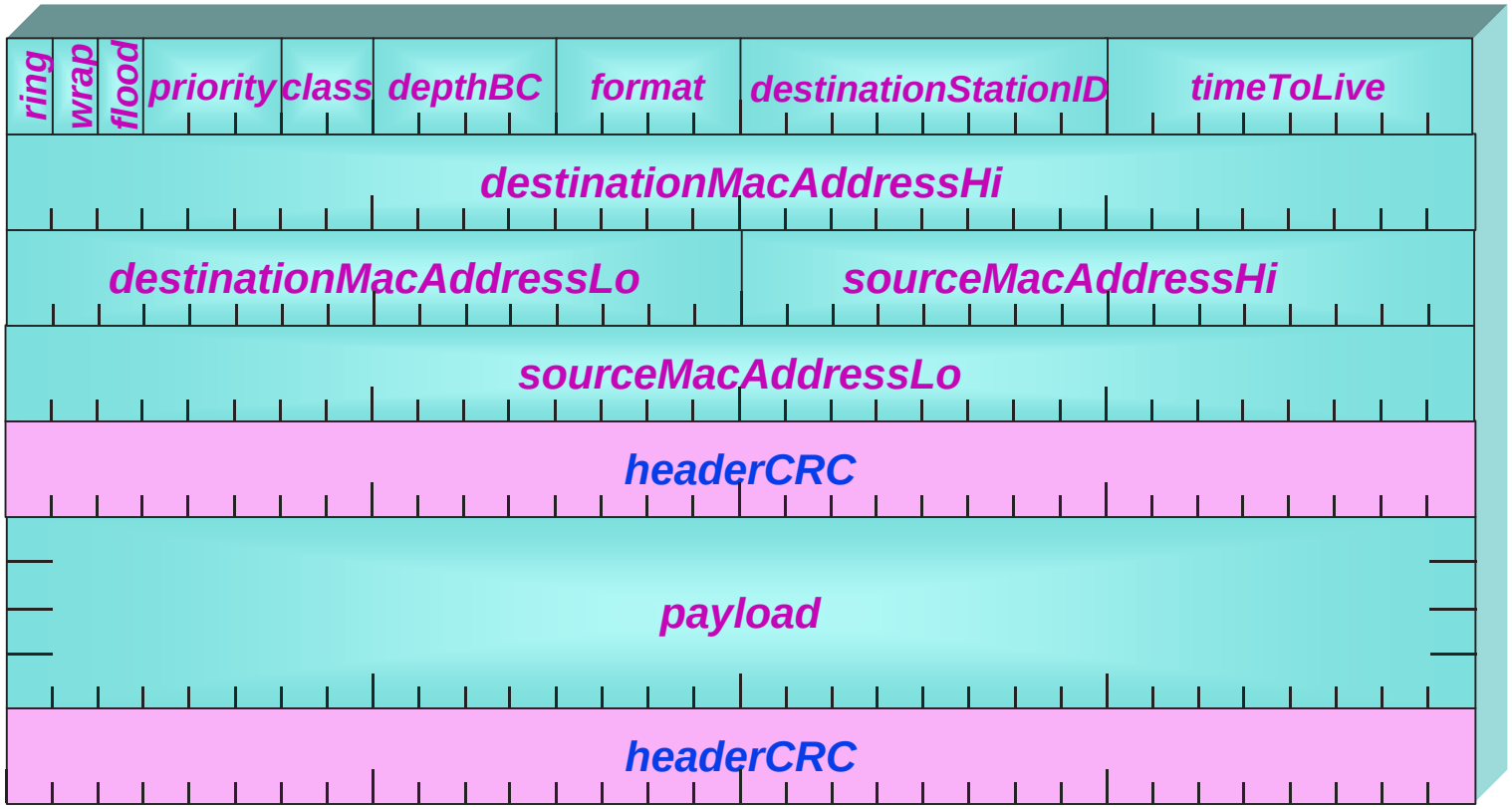
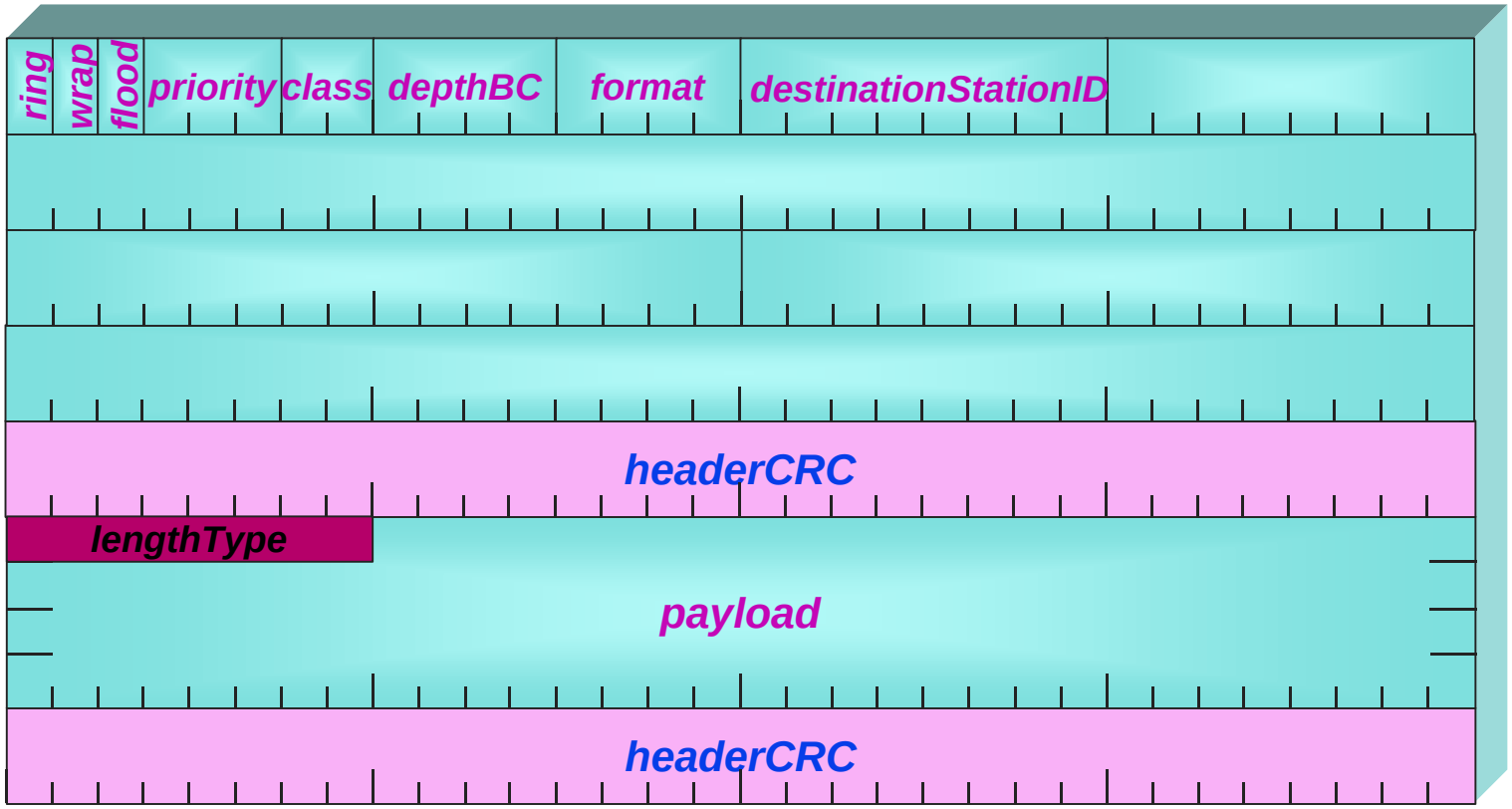
802.17 presentation

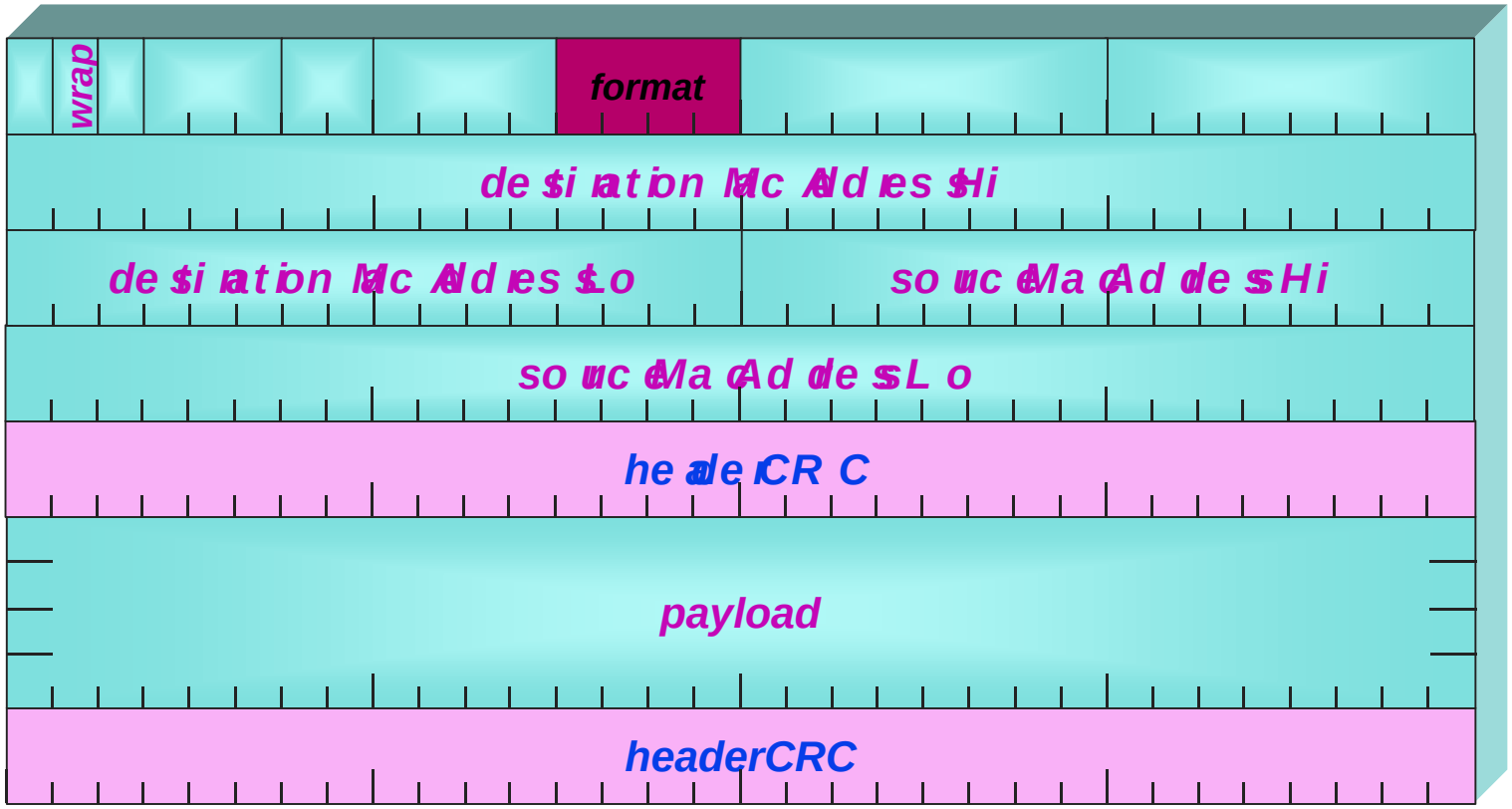
- Prepared for 802.17, November 2001
- Dr. David V. James
  Chief Architect
  Network Processing Solutions
  Data Communications Division
  110 Nortech Parkway
  San Jose, CA 95134-2307
  Tel: +1.408.942.2010
  Fax: +1.408.942.2099
  Base: dvj@alum.mit.edu
  Work: djz@cypress.com

RPR Frame Format



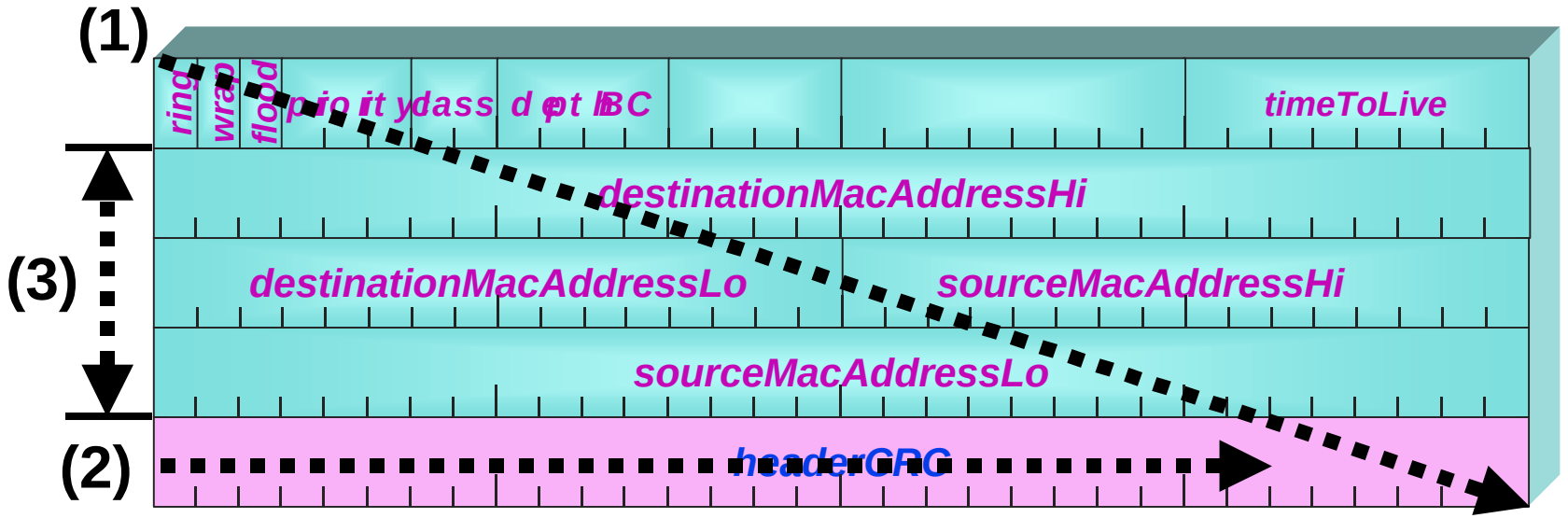

ring
wrap
flood
priority
class
depthBC
format
destinationStationID
headerCRC
lengthType
payload
headerCRC

Control Frame



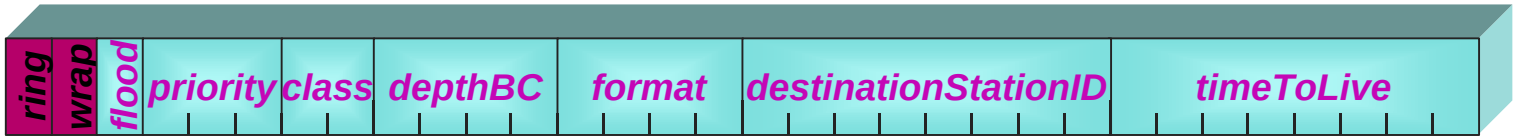
Control Field Functionality

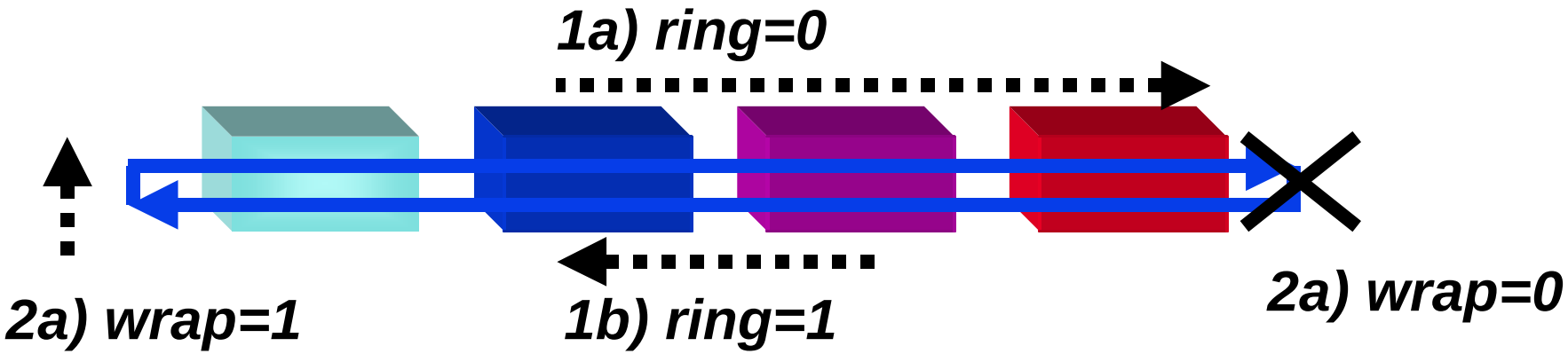
RPR Frame Format



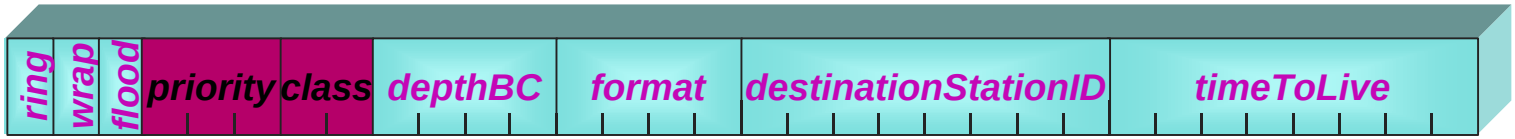
1) 32-bit aligned
2) 32-bit checksum
3) Global MAC addresses (not local)

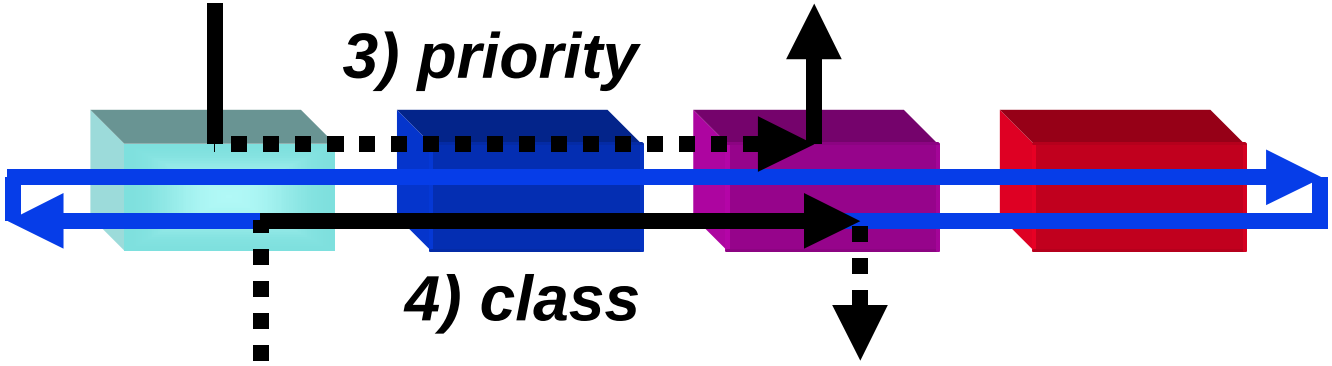
RPR Leader Format



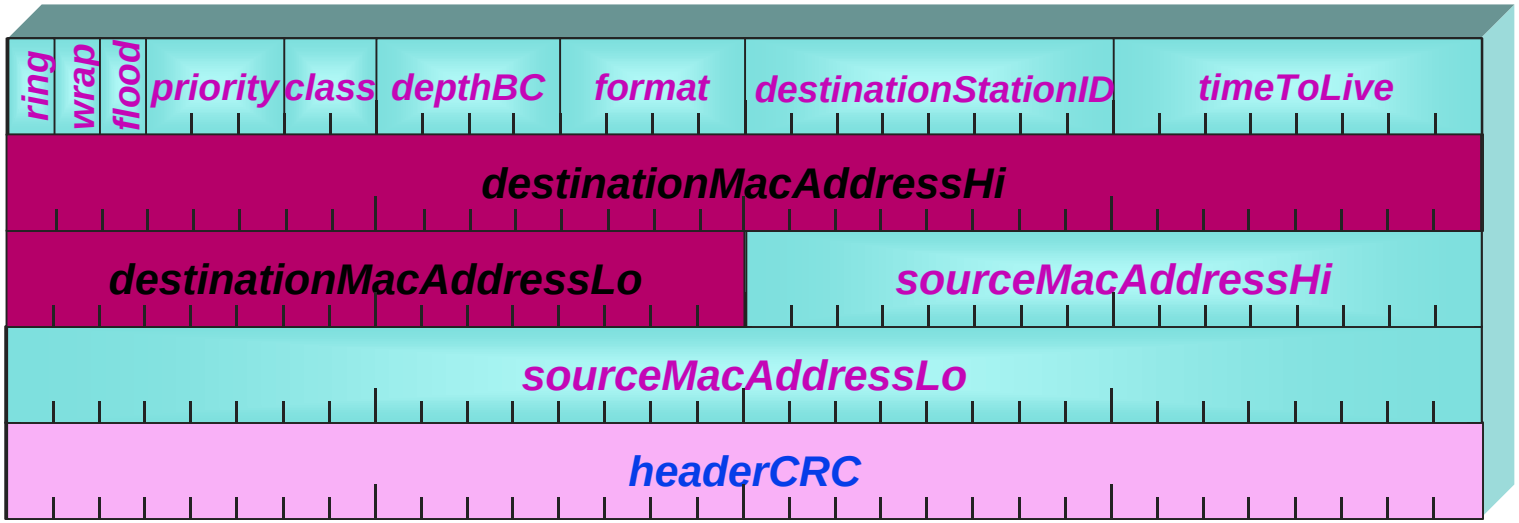


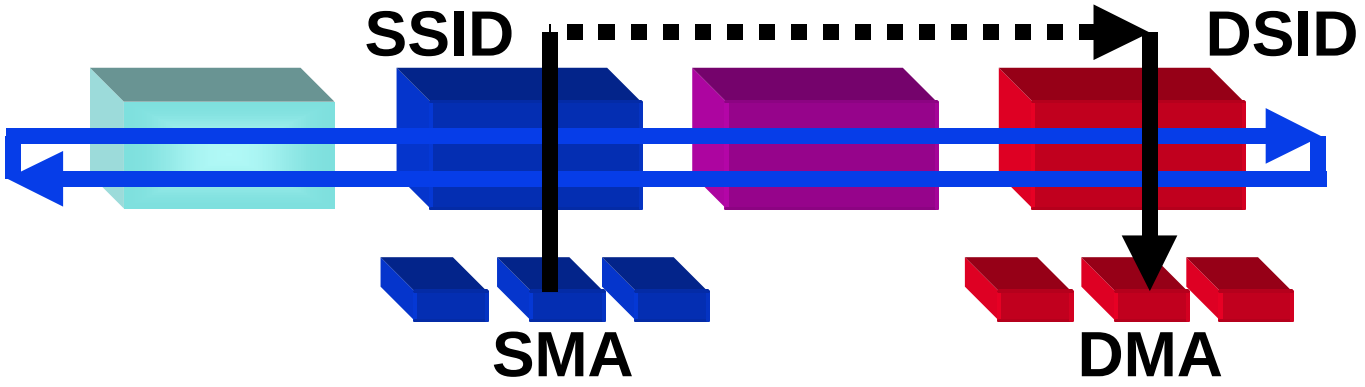
RPR Leader Format





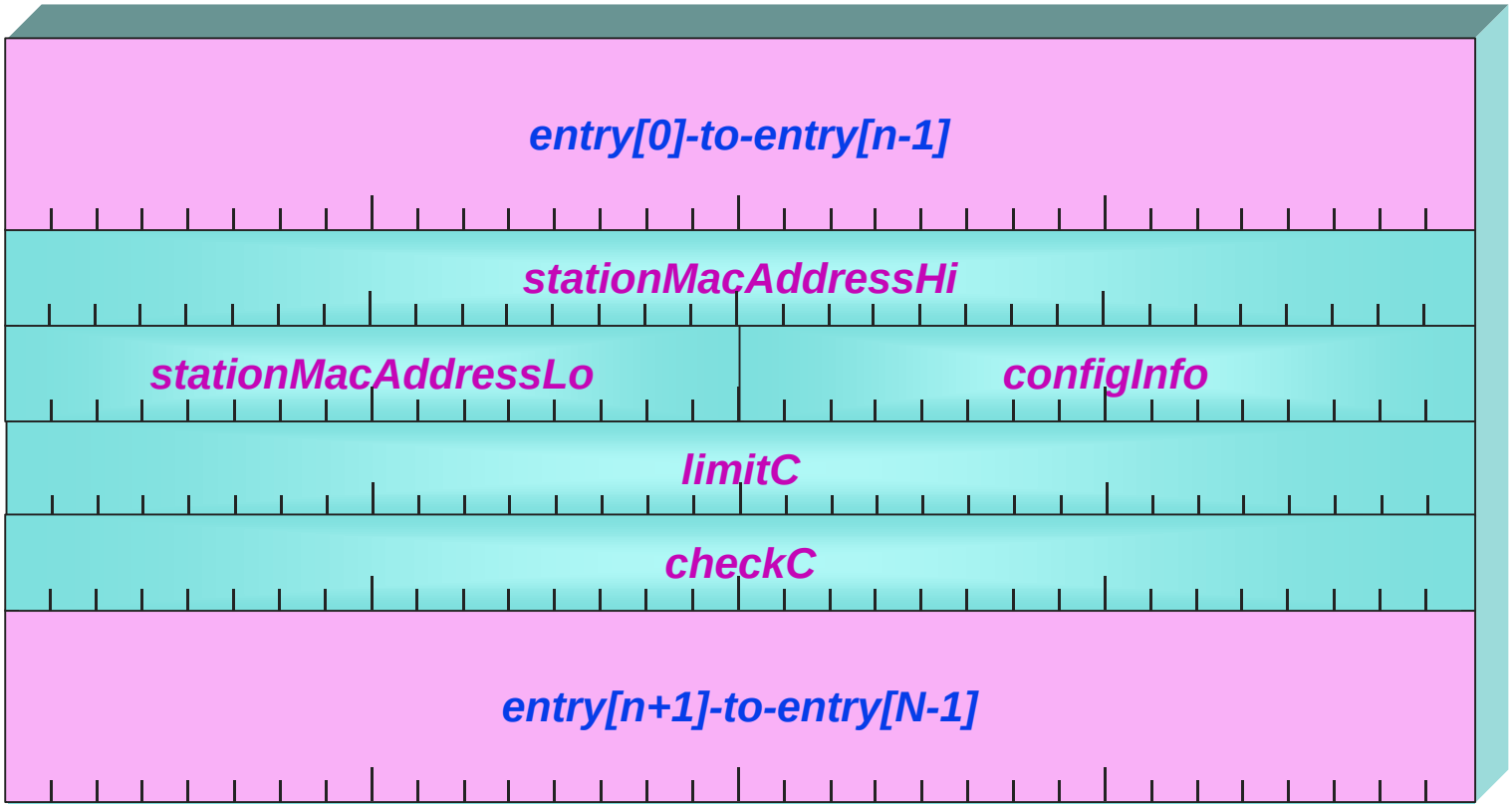
Ethernet Bridging



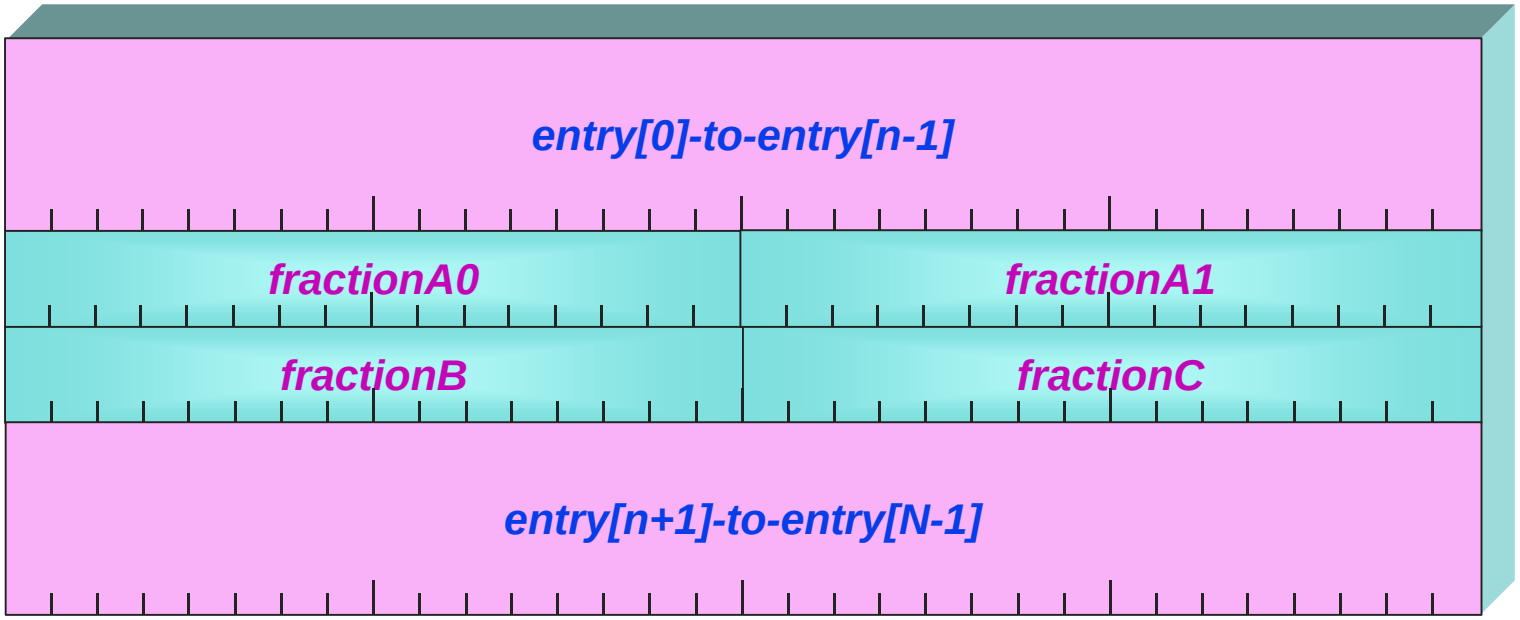


Control Frame Formats

Discovery Frame Format



Survey Frame Format

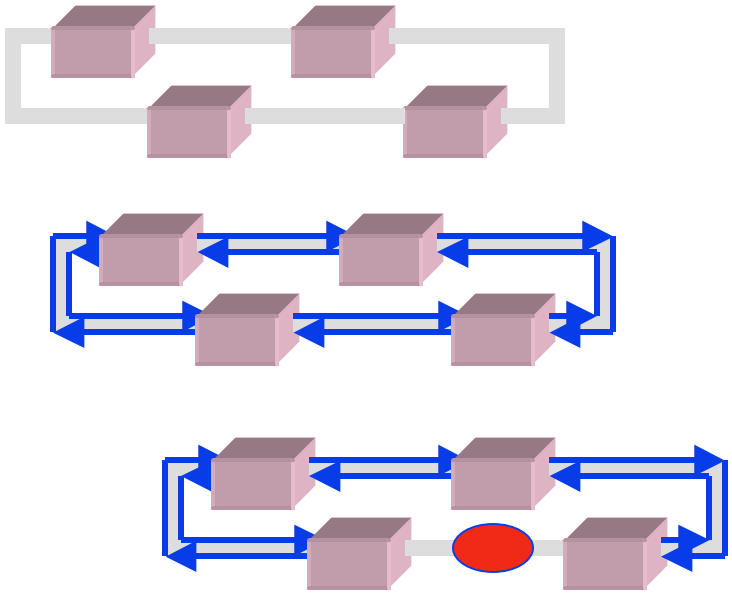


Format Issues

- Wrap: static versus dynamic
- Structural differences:
  - Alignment: 32-bit versus *16-bit
  - CRC coverage: 32-bit versus *16-bit
- Ethernet-type: payload vs *header
- Priority and class: distinct vs *merged
- Local addressing:
  - SSID= TTL, destination= DSID
  - *DSID= TTL, SSID= ????
- Class-A flow-control: embedded vs distinct

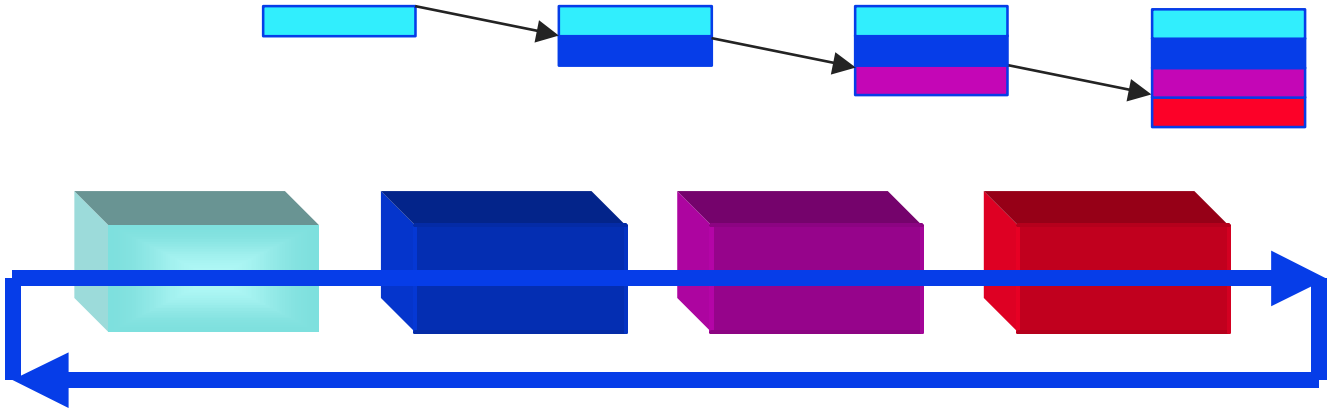
Discovery Sequencing

Supported topologies



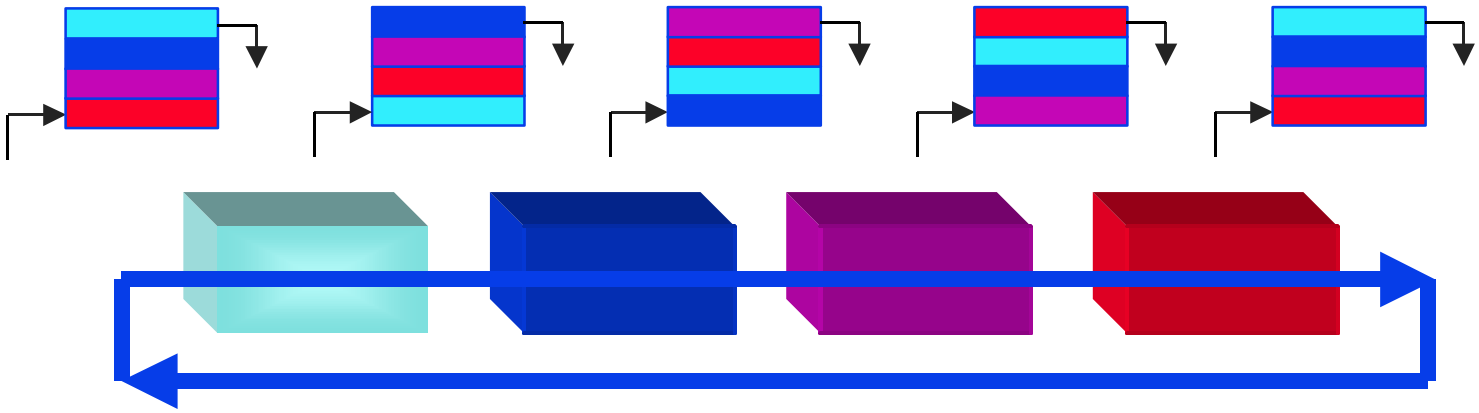
- A physical ring
- Dual ringlets
- Duplex ringlet

Topology collection



- Append your macAddress & info (no duplicate copies present…)

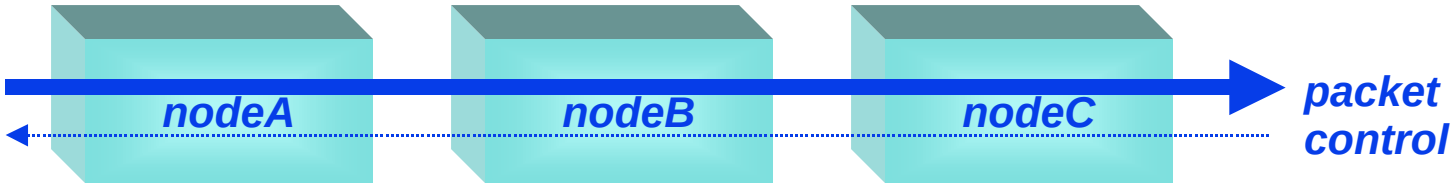
Topology Discovery



- Strip up-to existing macAddress (inclusive)
- Postpend your macAddress & information

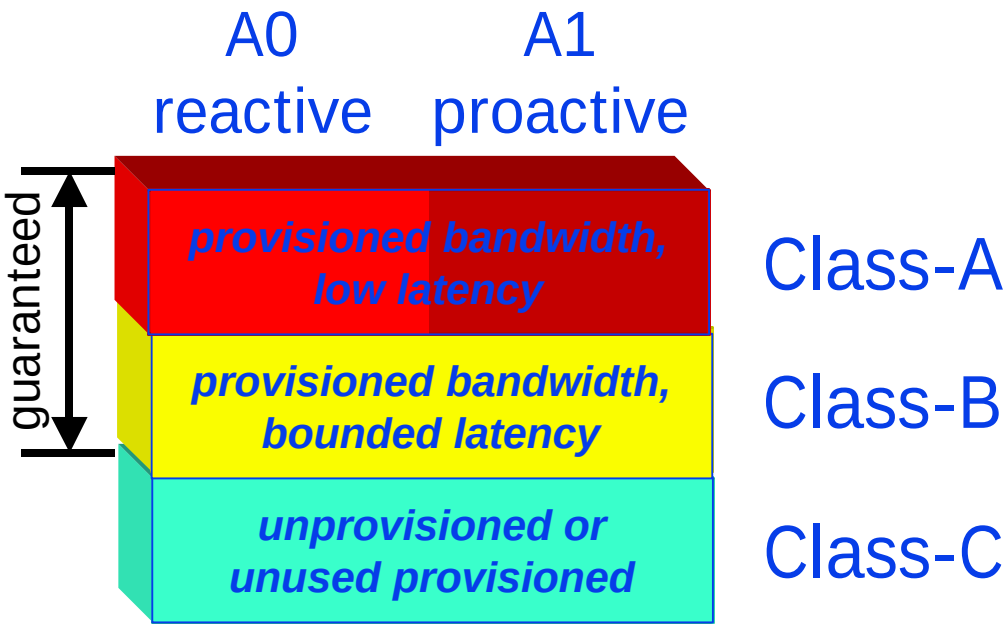
Flow control

Opposing arbitration

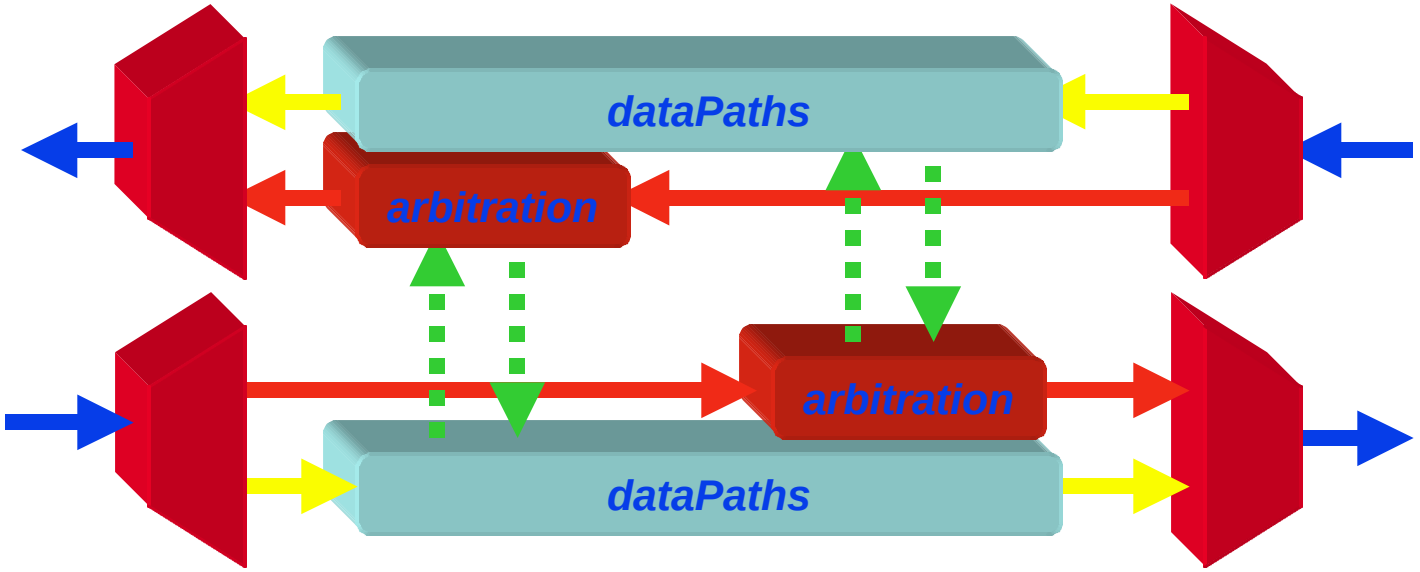


- Data packets flow in one direction
- Arbitration control flows in the other*

Arbitration classes

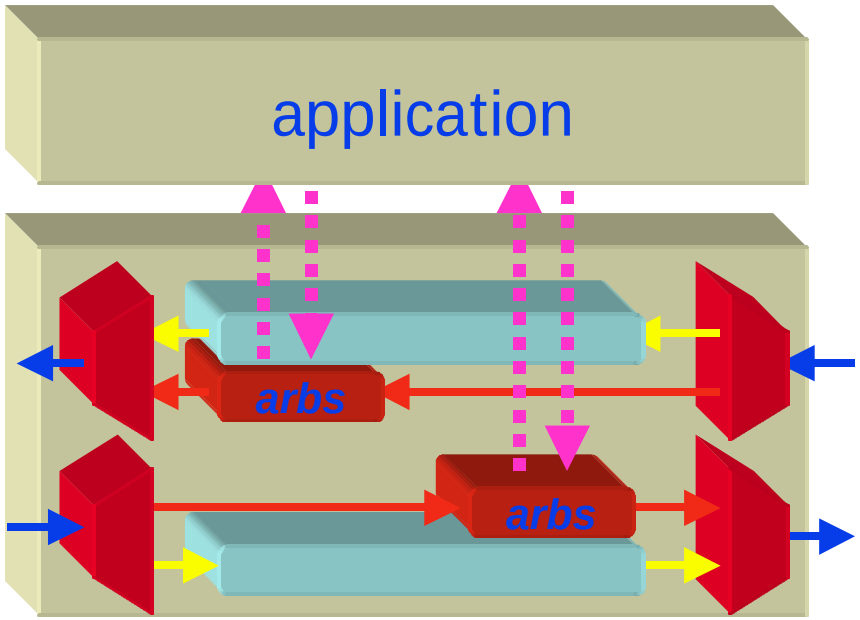


Internal MAC arbitration signals



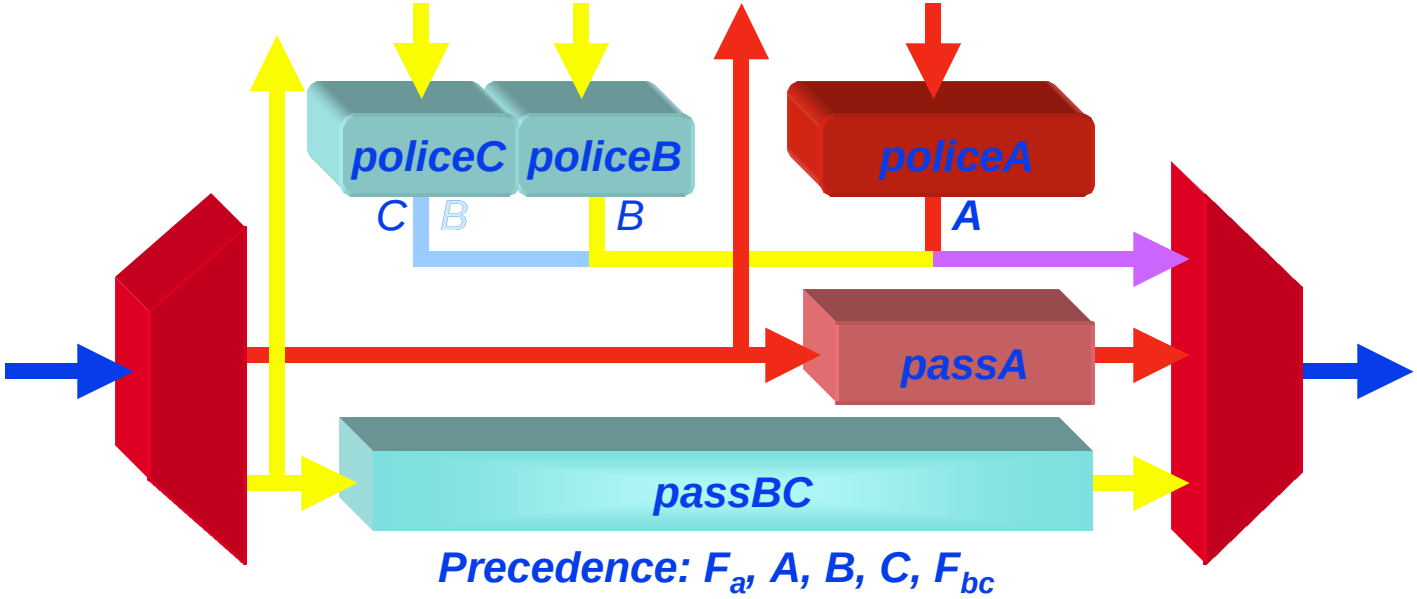
- Arbitration affects opposing run
- My congestion affects upstream node
- Downstream congestion affects me

External MAC arbitration signals



- MAC receives information
  - MAC FIFOs are $$, latency++, inflexible
- Application receives information
  - Allows reordering and run selection

Arbitration related components



- Distinct class-A & class-B/C paths
- Load dependent policing

Class-A flow control (proactive and reactive)

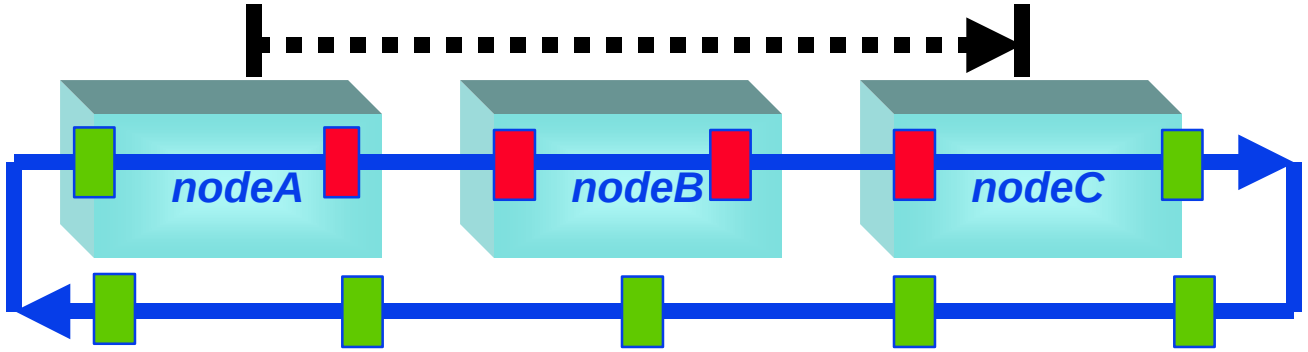
Class-A flow control

- **Proactive**
  Minimal (nonexistent?) passBC transit buffer
  Less available bandwidth
  Each station maintains constant classAp traffic

- **Reactive**
  Significant passBC transit buffer
  Full bandwidth utilization
  Each station responds/regenerates throttle messages

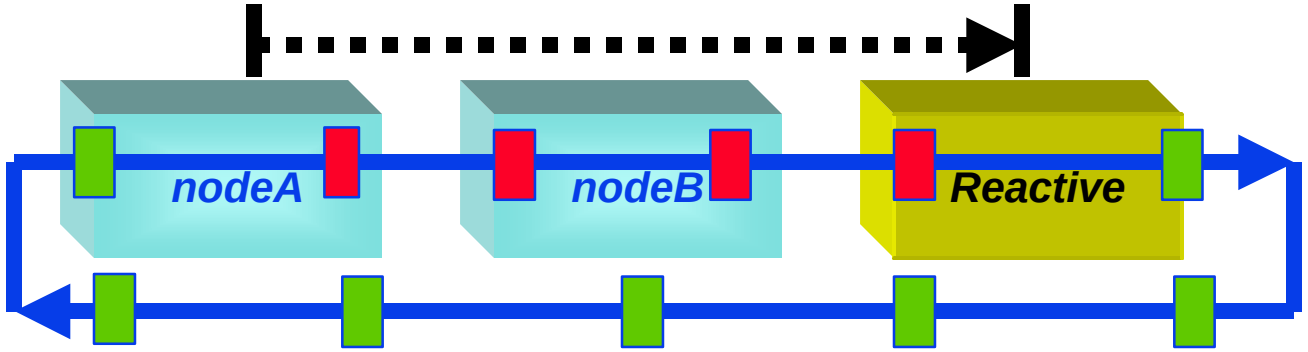
- **Interoperable?**
  This is a bandwidth vs memory $$ tradeoff

Proactive class-A partitions



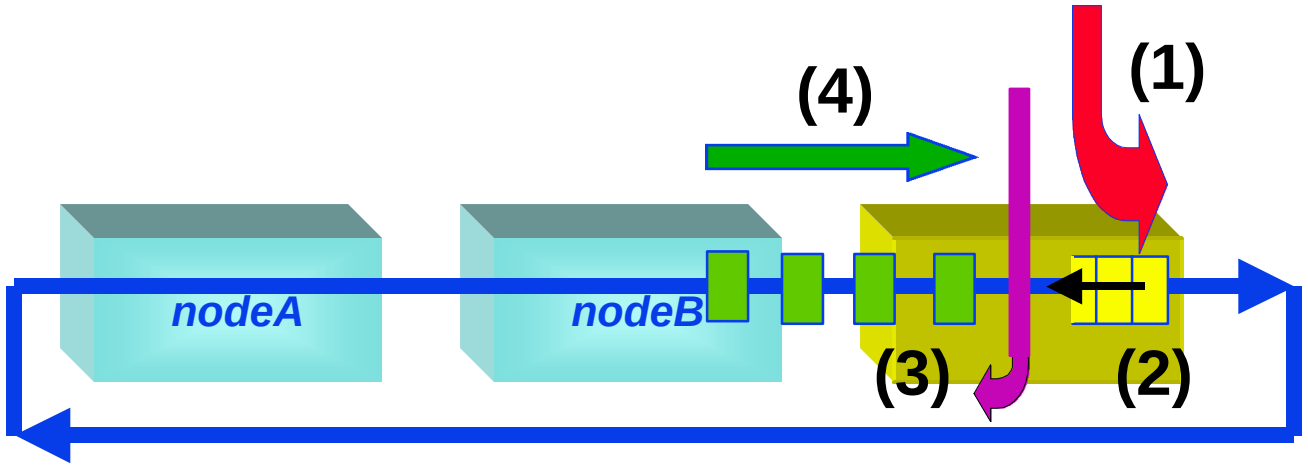
- Data packets go source-to-destination
- Residue returns destination-to-source to provide subsistence for transmissions

Proactive class-A compatibility options



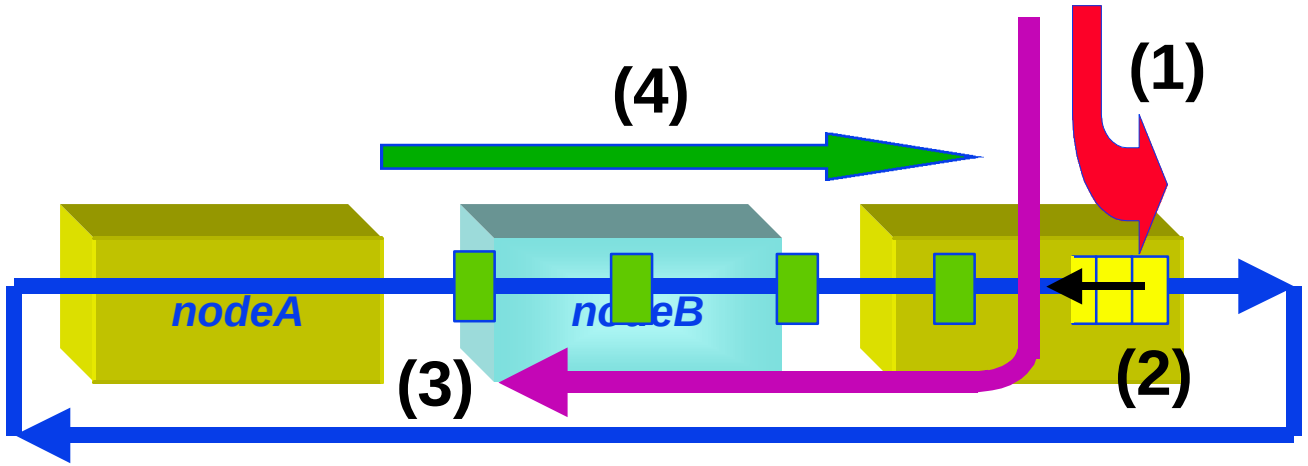
- Reactive node trickles class-A bandwidth
- Reactive node recycles class-A bandwidth class-A => class-A', thus preserving BW

Reactive class-A control



- Transmission of packets causes
- Backup of passBC FIFO that
- Returns flow-control information that
- Provides consumable idle packets

Reactive class-A compatibility



- Flow control passes upstream
- Proactive stations pass these indications

Topology discovery

Frame interchanges

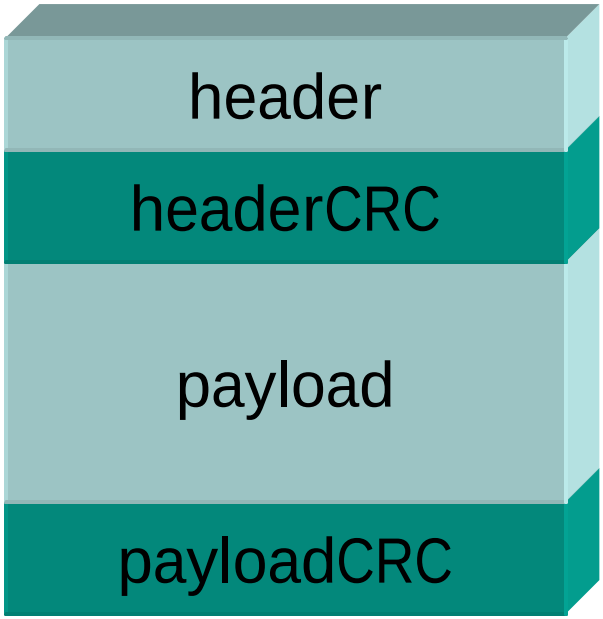
- Triggered on state change
- Triggered on state change
- Also sent periodically
  - Automatic fault recovery
  - Piggyback on heartbeat
- Also distributes stationID addresses
  - Previous: derived from topology and EUI-48 info
  - Bit map supportive “reclaiming” precedence
- Robust!
  - Context-less behavior (update rate only)
  - No addressing or timeouts required

CRC processing

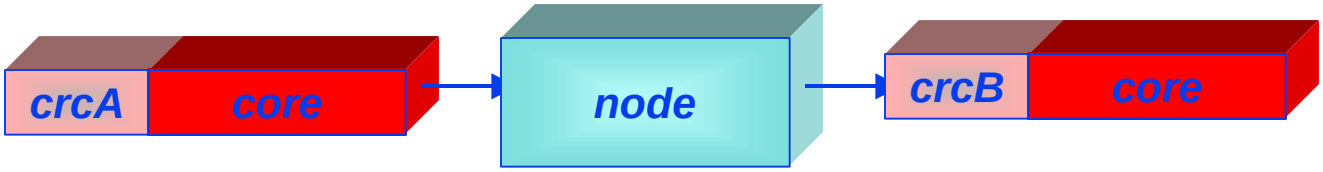
CRC processing

- Store&forward/Cut-through agnostic
- Invalid data is effectively discarded
  - store-and-forward discards
  - cut-through stomps the CRC
- Maximize error-logging accuracy
  - Separate header&data CRCs
  - “most” corruptions hit the data

Separate header and data CRCs

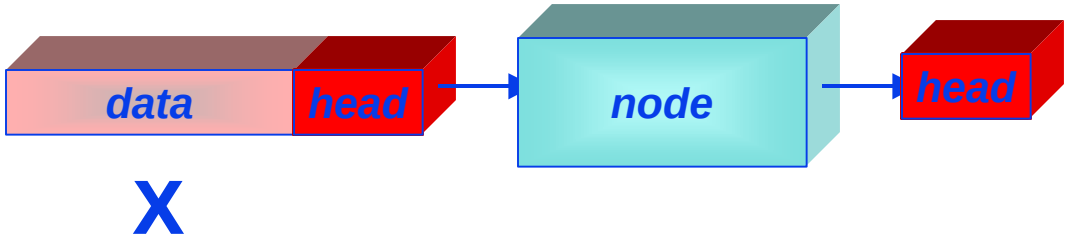


Cut-through CRCs

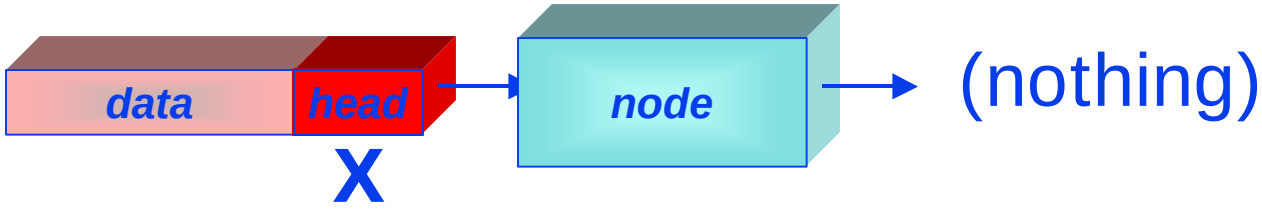


- Corrupted packet remains corrupted
- Error logged when first detected
- if (crcA!=crc) {
  errorCount+= (crcA!=crc^STOMP);
  crcB= crc^STOMP;
  }

Distinct CRCs reduces discards

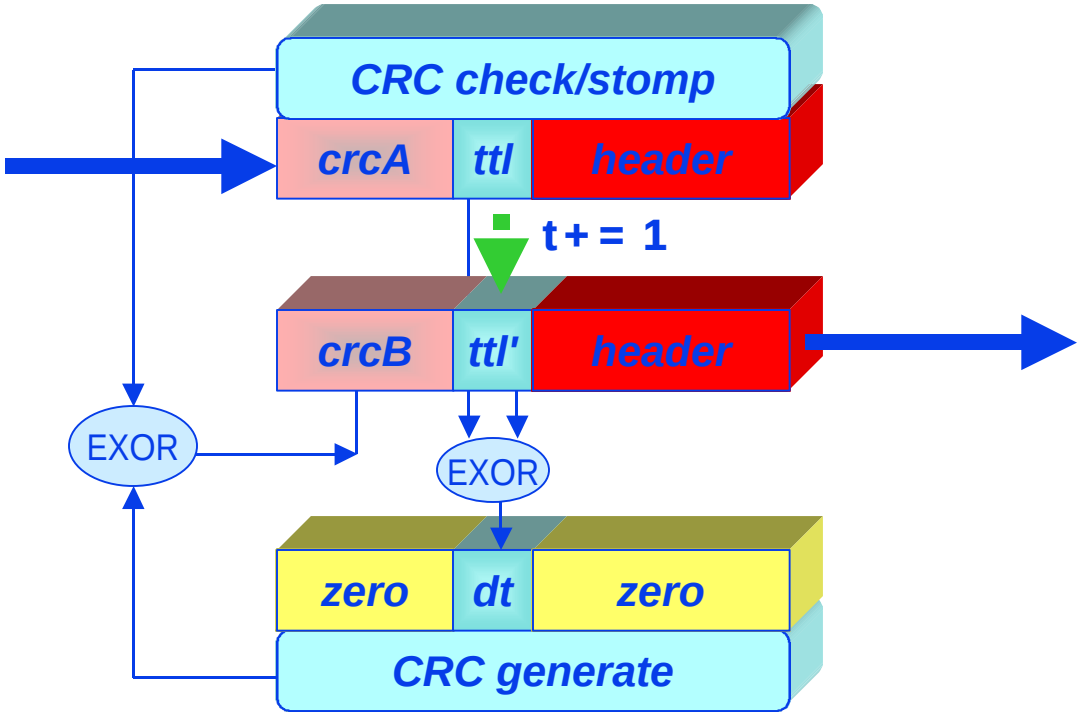


- Discard the corrupted data



- Discard the corrupted packet

End-to-end CRC protected TTL

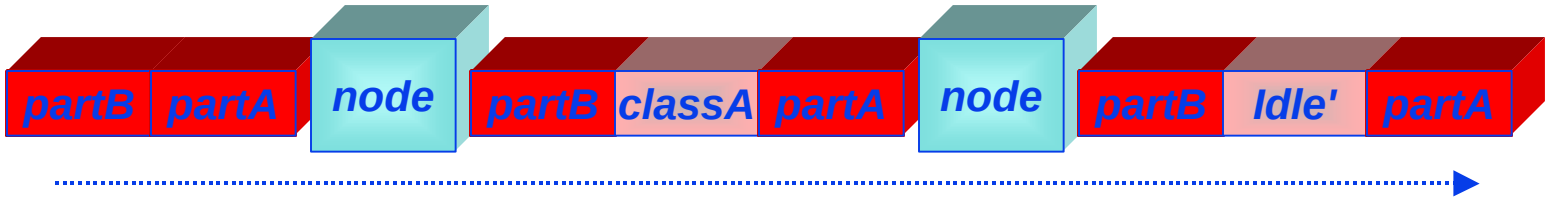


Pre-emption
(a physical layer decision)

Pre-emption

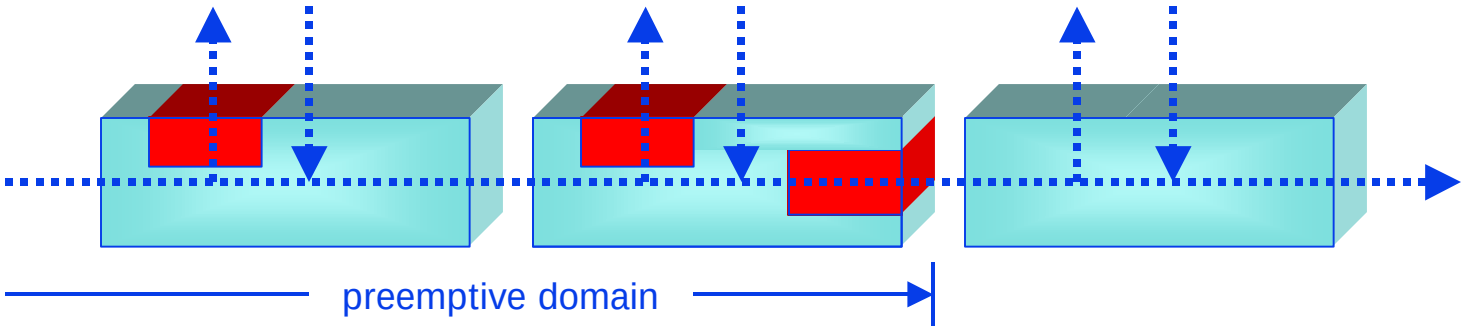
- Suspend class-B/C for class-A packet
- Only one level is sufficient
  - class-A is the latency critical traffic
  - more levels complicate hardware
- Physical layer dependent
  - marginal for high BW & small packets
  - distinctive “suspend” symbol required

Pre-emption fragments



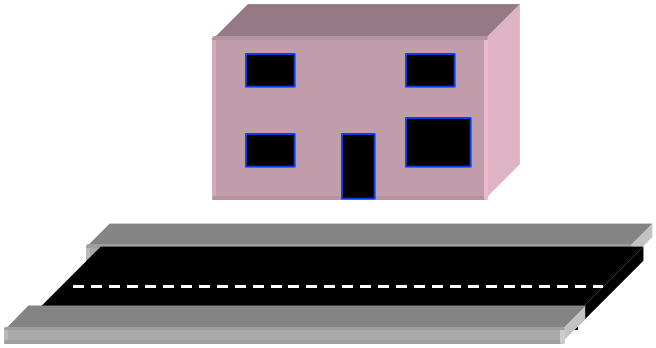
- Packets can be suspended
- The class-A packet can be stripped
  - egress queues are store&forward
  - distinctive idle markers needed

Pre-emption compatibility



- Pre-emption mandates egress S&F
- Simplistic node has no such S&F
- Interoperability burden on elegant
  - boundary node has S&F bypass
  - cut-through in preemptive domain

Limits of scalability



- Geosynchronous
  - Terrestrial
    - The metro area
      - To the curb
        - To the home

Lessons of the past…

- Flow control mandates 2-out-of-3
  - Low latency transmissions
  - Fair bandwidth allocation
  - High bandwidth utilization
- Feedback control systems
  - Low latency signaling
  - Control can pass class-B/C packets
  - Separate class-A queue is utilized
- Other observations
  - Local control => global perversions
  - Fairness is inherently “approximate”
  - Strange beating sequences DO OCCUR

Allowed transmissions

	warnings		transmissions		
	LO	HI	none	LO	HI
≥3/4	send	send	A,F	A,F	A,F
≥1/2	send	pass	A,F	A,F	A
≥1/4	pass	--	A,B,C_b,F	A,B	
≥0	--	--	A,B,C_b,C_c,F		

Arbitration summary

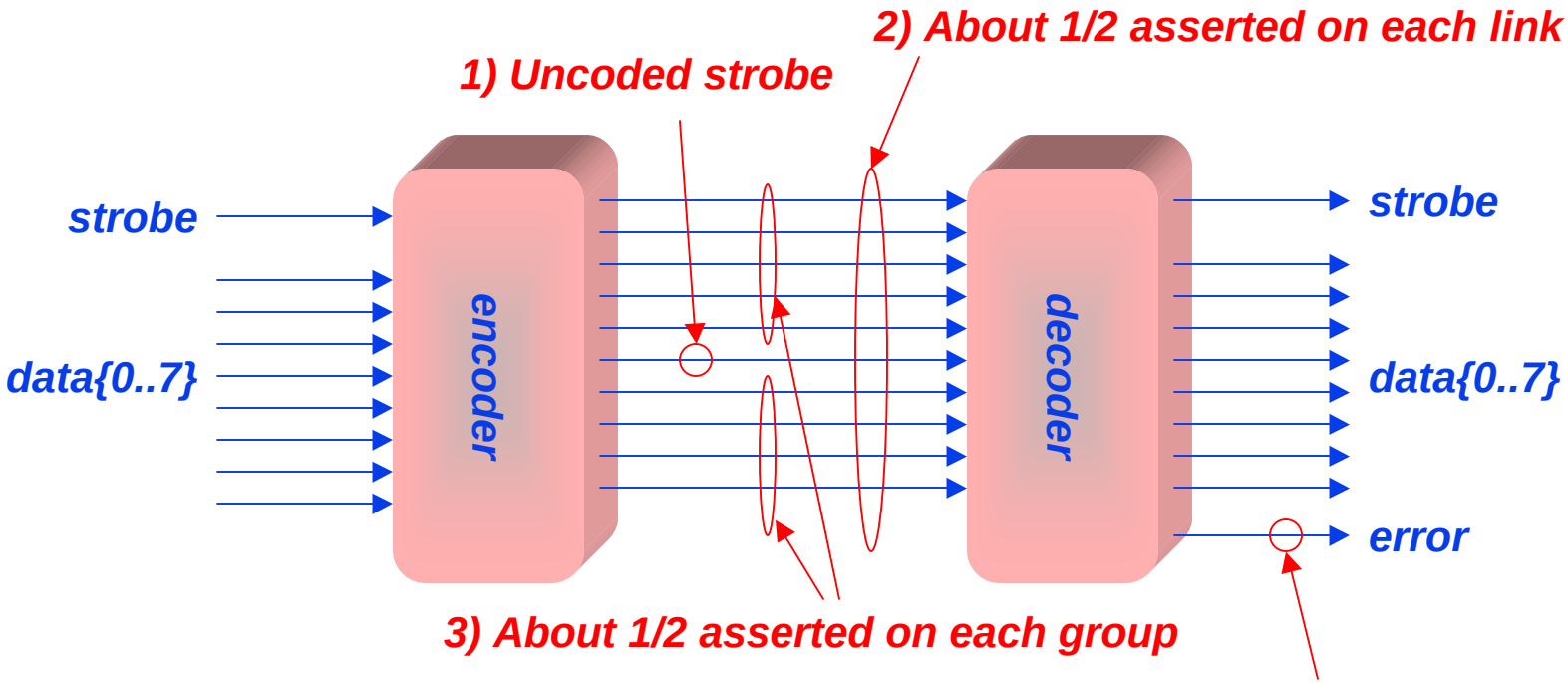
- Dual levels
  - Class-A, pre-emptive low latency
  - Class-B, less latency sensitive
- Jumbo frames
  - Affect asynchronous latencies
  - NO IMPACT on synchronous latency
- Cut-through vs store-and-forward
  - Either should be allowed
  - Light-load latency DOES matter

MAC interface links

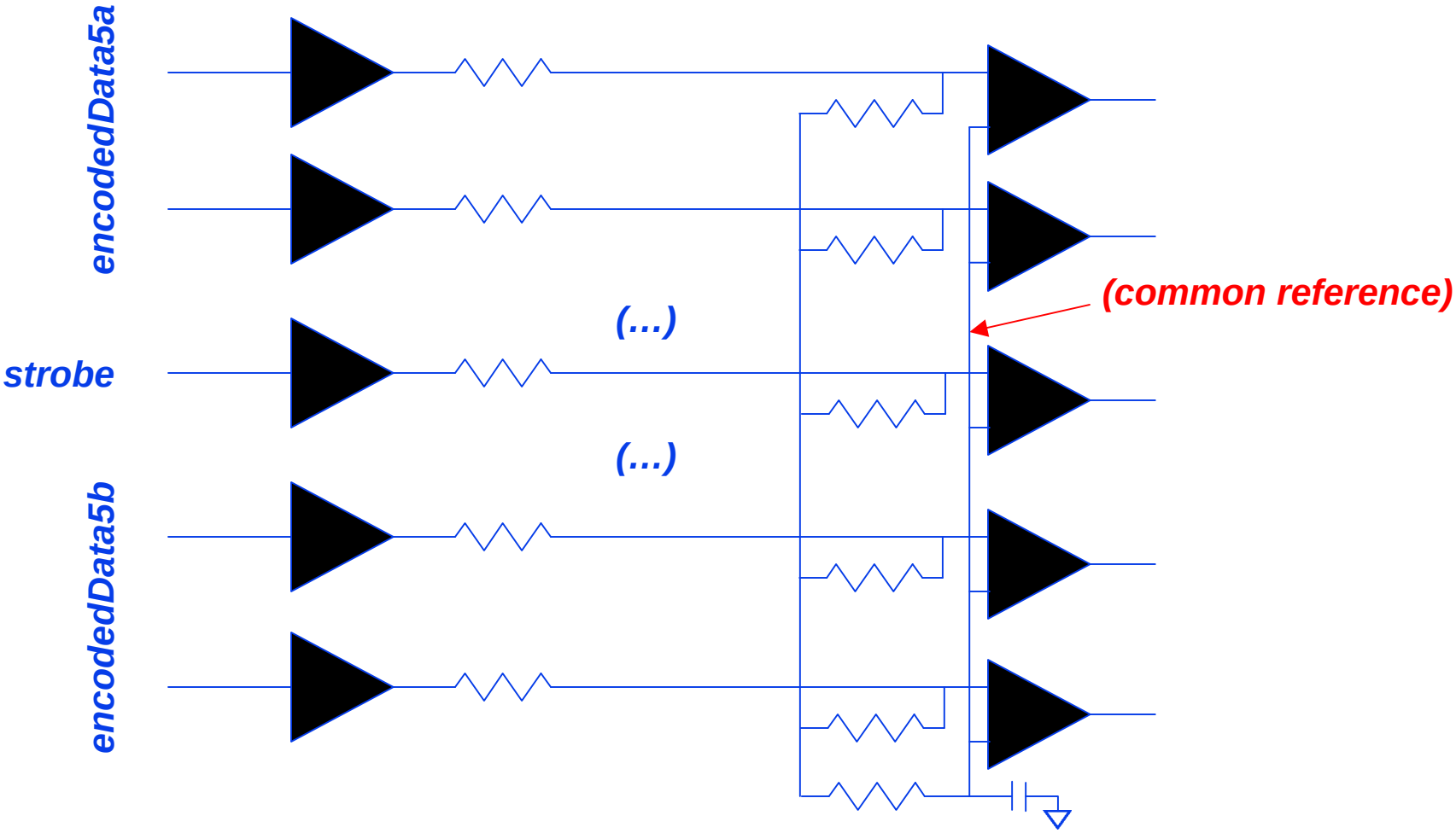
LiteLink features

- Byte-wide 10Gb/s link (Gb/pin/s)
  - 11 pins total for clock, control, and data
- SLVS-400 transmitter, matched impedance
  - Technology independent 0.8V driver supply
  - .2V and .6V low and high signal levels
- Differential receiver, matched impedance
  - Near DC encoding (5/11 or 6/11 ones)
  - Termination derived reference voltage
- Source synchronous, DDR clock

Byte-wide coding properties



LiteLink signaling



Common features

Common features

- +Separate header and payload CRCs
- +Virtual output queues for efficient spatial reuse
- +Proactive&reactive class-A traffic options
- +Weighted fairness
- +Three fairness classes but distinct naming
  high/medium/low vs A/B/C
- +Node count: >=63, with a desire for 256
  (TTL w/wrap is much simpler if <=127)
- +Wrap and steering supported

Similar themes

- +Duplex queues: Gandolf & DVJ
- +Cumulative discovery: Gandolf & DVJ
- +Steering/wrapping specified on per-packet basis
- #DVJ: Client-to-MAC physical interface
- #DVJ: Clock differences (elasticity buffer mgmnt)
- #DVJ: Time-of-day (stratum check)
- #DVJ: Brandwidth reservation management (for consistent provisioning)
- #DVJ: CRC-32 formats (MAC assumes only one?)

Contending mechanisms

- -More than duplex (x2) ringlets
  DVJ&Gandolf: x2 duplex ONLY
  Alladin: xN if not found to be “overly” complex
- -Flow control (B and C)
- -Frame format fields
  - Presence or absence of stationID fields
  - "Questionable" value fields
  - header vs payload, for type & CID
- Discovery
  DVJ&Gandolf: Cumulative discovery
  Alladin: Multistep