

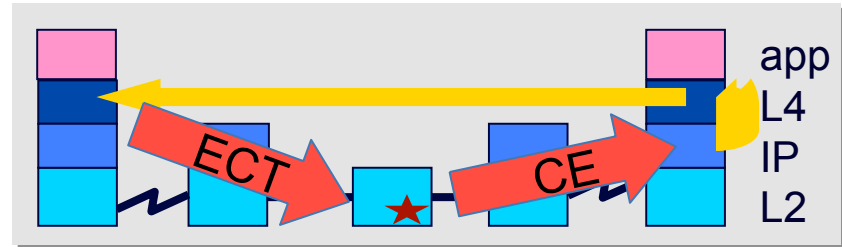
# TRILL ECN Support

draft-eastlake-trill-ecn-support-01.txt

Donald E. Eastlake, 3<sup>rd</sup>  
[d3e3e3@gmail.com](mailto:d3e3e3@gmail.com)

Bob Briscoe  
[ietf@bobbriscoe.net](mailto:ietf@bobbriscoe.net)

# ECN Background



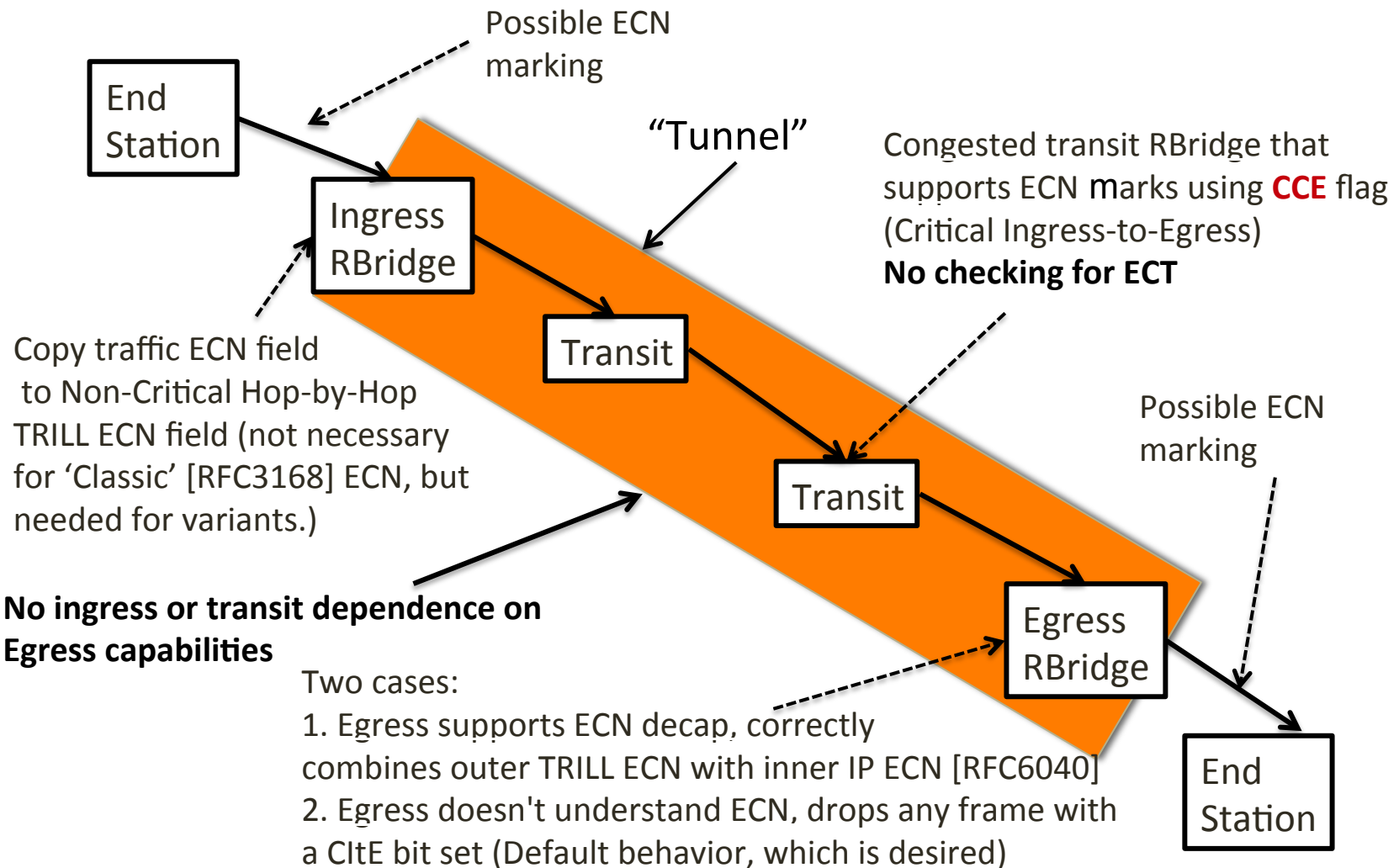
- ECN propagates
  - “ECN-capable transport” (ECT) down
  - Congestion Experienced (CE) up
- ECT is necessary for incremental deployment
  - See IP ECN codepoints table (right)

IP-ECN codepoint	value	meaning
Not-ECT	00	Not ECN-capable transport
ECT(0)	10	ECN-Capable Transport
ECT(1)	01	ECN-Capable Transport
CE	11	Congestion Experienced ('marked')

- Similar incremental deployment problem for TRILL
  - if legacy **egress** does not understand ECN
  - will not propagate upward to forwarded IP inner header
  - would black-hole congestion signals



# Defer mark / drop decision to egress



# Recap: ECN tunnelling rules at egress [RFC6040]

incoming inner	Arriving TRILL 3-bit ECN codepoint			
	Not-ECT	ECT(0)	ECT(1)	CE
Not-ECT	Not-ECT	Not-ECT	Not-ECT	drop
ECT(0)	ECT(0)	ECT(0)	ECT(1)	CE
ECT(1)	ECT(1)	ECT(1)	ECT(1)	CE
CE	CE	CE	CE	CE
Outgoing header				

TRILL egress same as [RFC6040] but 3 TRILL ECN bits. So map 3 bits to the 4 codepoints as shown in table:

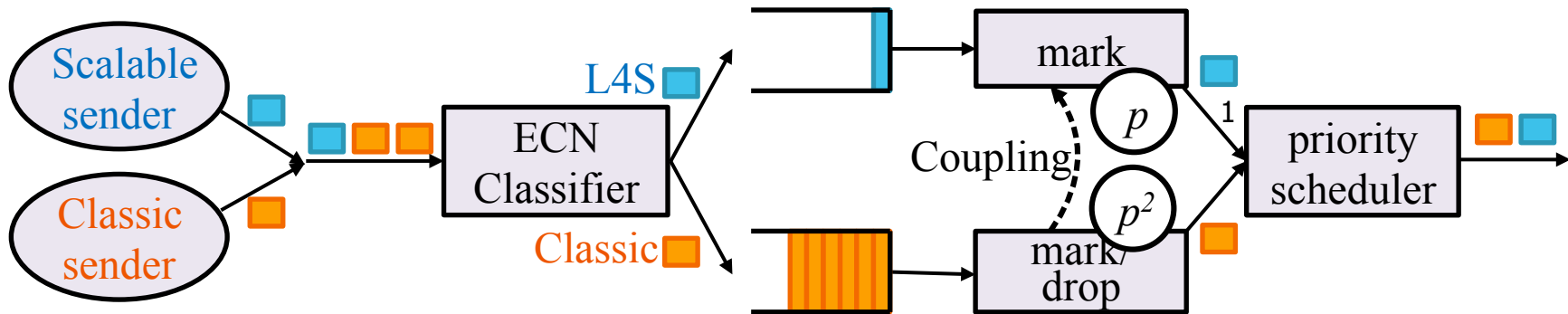
NCHbH TRILL ECN	CitE CCE	Arriving TRILL 3-bit ECN
00	0	Not-ECT
10	0	ECT(0)
01	0	ECT(1)
11	0	CE
00	1	CE
10	1	CE
01	1	CE
11	1	CE

# Changes in Draft -00 > -01

- Last IETF: presented 3 possible solutions
  - draft-00 wrote up solution #2
  - re-written to specify chosen solution: #3
- Renamed two CE fields to:
  - Non-Critical Congestion Experienced
  - Critical Congestion Experienced
- Added section on support for ECN variants
  - pre-congestion notification (PCN)
  - L4S (successful BoF on Tuesday)...

# Adding support to TRILL for Low Latency Low Loss Scalable throughput (L4S)

- for background on L4S see:
  - draft-briscoe-aqm-duaq-coupled, draft-briscoe-tsvwg-ecn-l4s-id



- On transit TRILL RBridge classify on TRILL-ECN field, then

```
Classic queue:
if (p > max(random(), random() ) {
    mark(frame, CCE )
}
```

```
L4S queue:
if (p > random() ) {
    if (p' > random() ) mark(frame, CCE )
    else mark(frame, NCCE )
}
```

- then deferring mark/drop decision to egress gives desired outcome
- without any L4S logic at the egress

# Next Steps

- Review the draft please
  - comprehensibility
  - Implementability
- WG adoption call



# End

Donald E. Eastlake, 3<sup>rd</sup>  
[d3e3e3@gmail.com](mailto:d3e3e3@gmail.com)

Bob Briscoe  
[ietf@bobbriscoe.net](mailto:ietf@bobbriscoe.net)