TRILL/802.1aq Activity

Ronald van der Pol
rvdp@sara.nl

Outline

- Scope/context of the TRILL/802.1aq activity
- Goal of the TRILL/802.1aq activity
- Proposed Timeline
- Basics of IETF TRILL
- Basics of IEEE 802.1aq
- Pros and cons of TRILL and 802.1aq
- Where to implement TRILL/802.1aq?
- Discussion

Scope/context

- TRILL/802.1aq concepts as such will (probably) not provide a full LHCONE network solution
- TRILL/802.1aq concepts may be useful in L2 Ethernet domains within LHCONE

Goal

- Investigate TRILL and IEEE 802.1aq and decide whether these protocols or concepts can be useful for LHCONE. (this is an R&D effort)
- Decide how TRILL/802.1aq concepts can be used in the *next generation* LHCONE service pilot

Timeline (1/2)

- End of Q1 2012: publish a document describing TRILL and 802.1aq.
  - What are the pros and cons?
  - What equipment or implementations are available?
- Rest of 2012: Setup a testbed with equipment that supports TRILL or 802.1aq or that uses their concepts.

Timeline (2/2)

- End of 2012: Make a decision:
  - Choose one of the protocols or concepts to explore further
  - *Or* decide this is not the way forward
  - *Or* decide we need more time to make a decision
- First Half of 2013: Implement concepts in a *next generation* LHCONE service pilot.

IETF TRILL

IETF TRILL Basics

- **TR**ansparent **I**nterconnection of **L**ots of **L**inks
- IETF Working Group since 2005 (internet area)
- Chairs: Erik Nordmark (Cisco Systems) & Donald Eastlake 3rd (Huawei Technologies)
- http://datatracker.ietf.org/wg/trill/charter

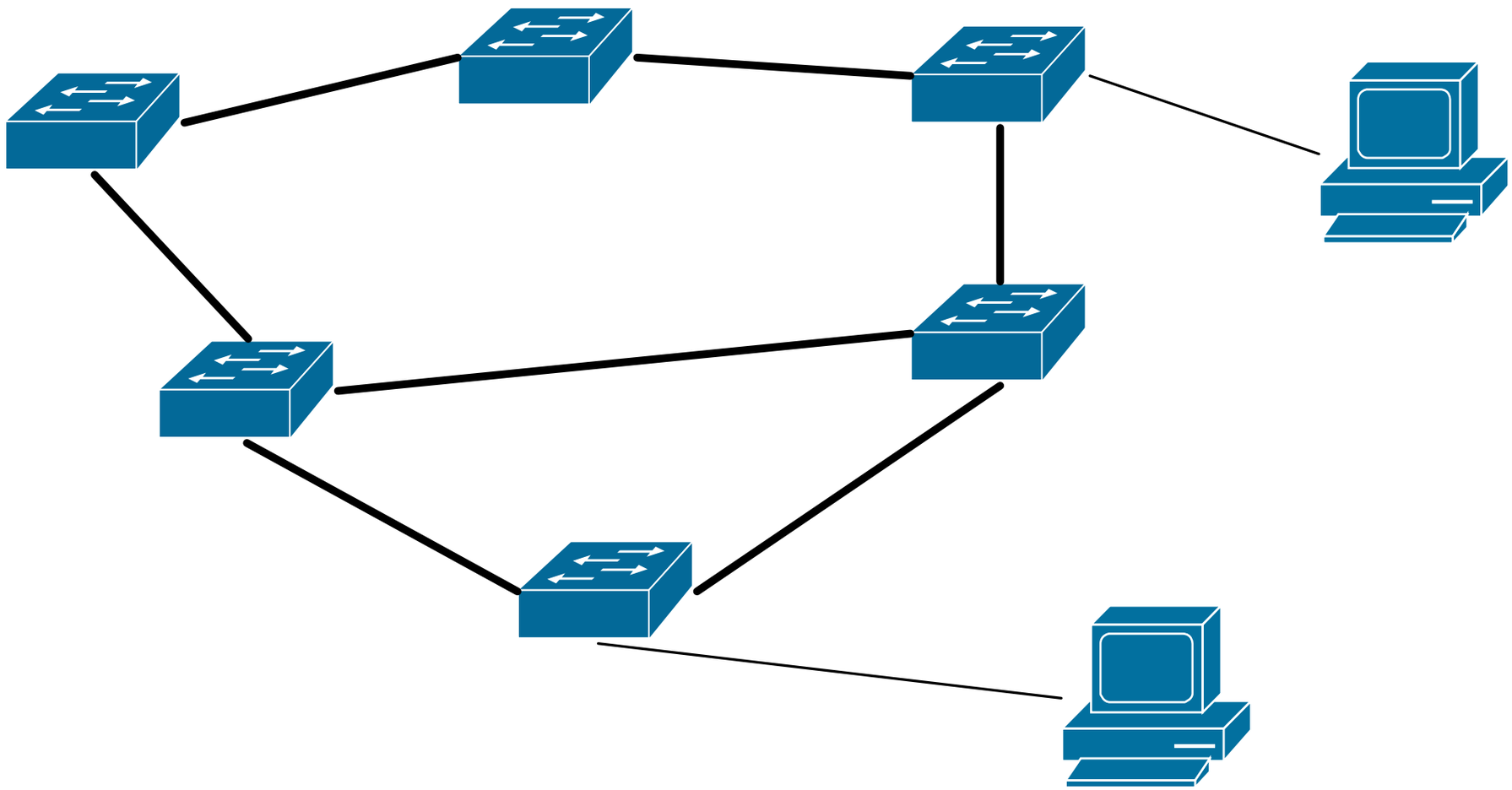
TRILL RFCs

- RFC 5556 - TRILL: Problem and Applicability Statement
- RFC 6325 - RBridges: Base Protocol Specification
- RFC 6327 - RBridges: Adjacency
- RFC 6439 - RBridges: Appointed Forwarders

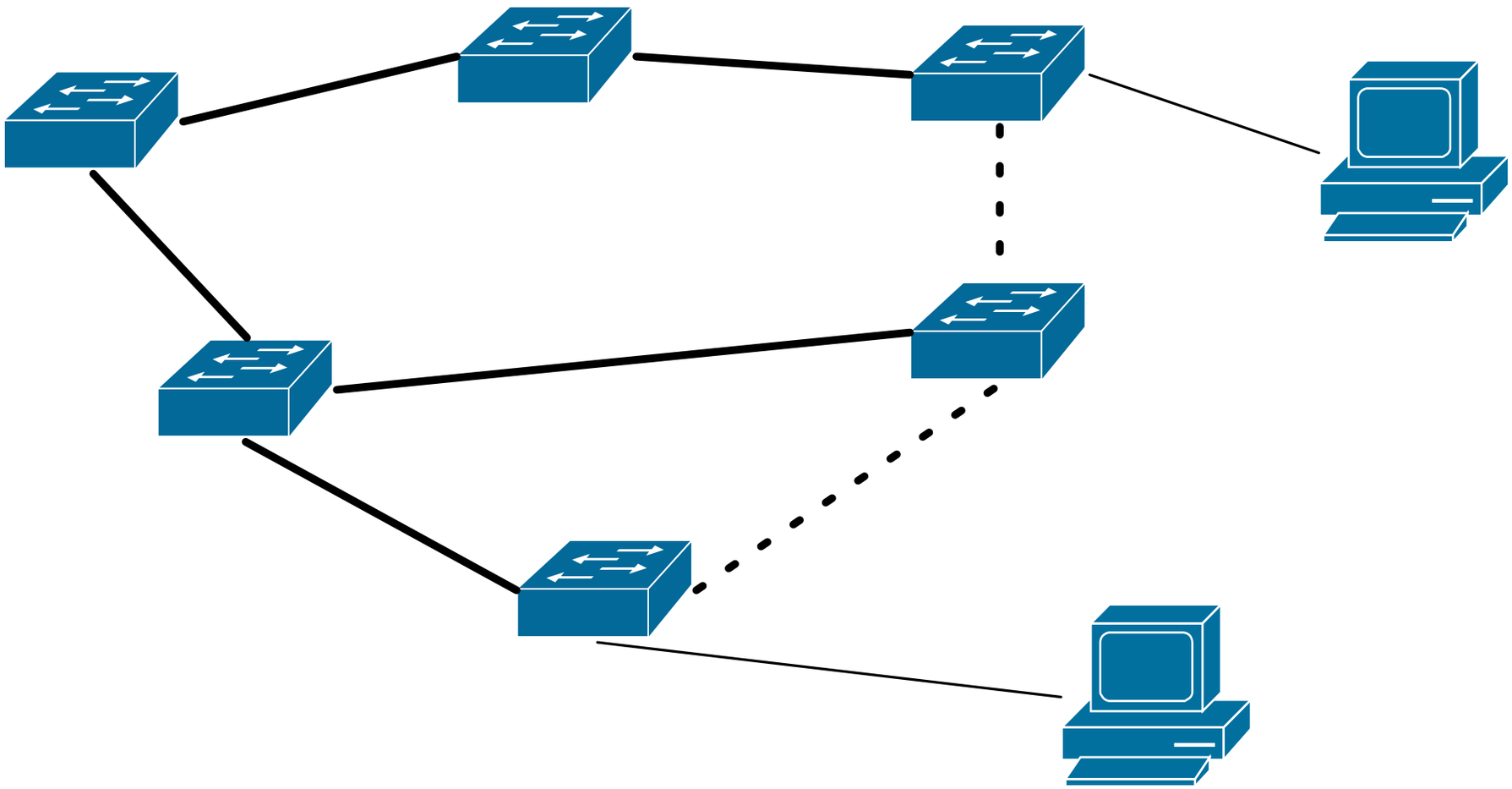
Problem Statement

- Spanning Tree Limitations
  - Traffic must follow tree instead of shortest path
    - No multipathing possible
  - Slow convergence after topology change
    - sometimes seconds to tens of seconds

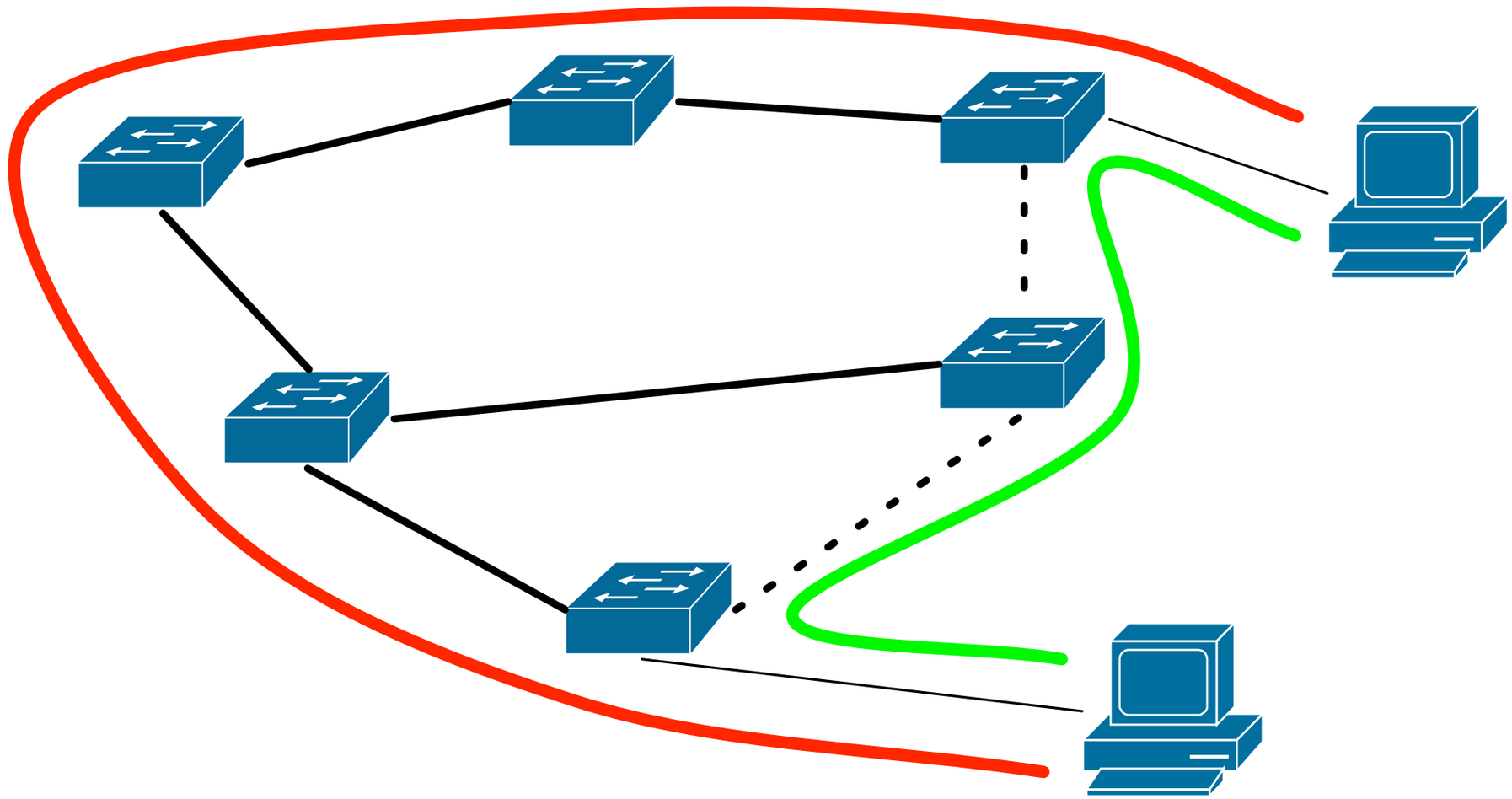
Topology with Loops



Spanning Tree



Sub-optimal Data



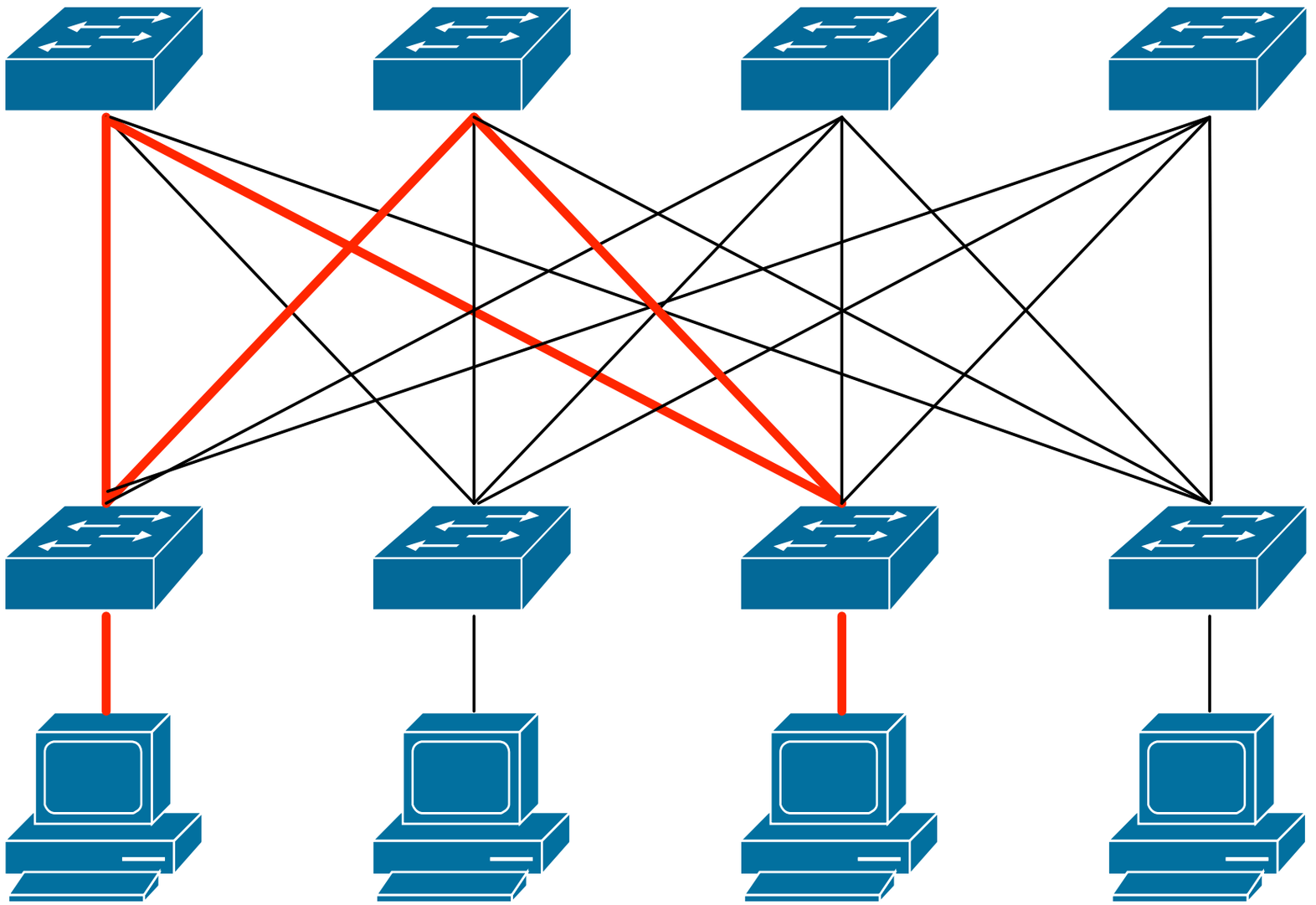
TRILL Solution

- L2 bridging using IS-IS link state routing
- Idea of Radia Perlman (inventor of the Spanning Tree Protocol)
- Building blocks of TRILL are Routing Bridges (RBridges)
- Encapsulate at ingress => route through domain using IS-IS => decapsulate at egress

RBridges

- RBridges provide shortest path forwarding between ingress and egress Rbridge
  - plug and play like switches
  - safe forwarding during temporary loops
  - multipathing support

Multipathing



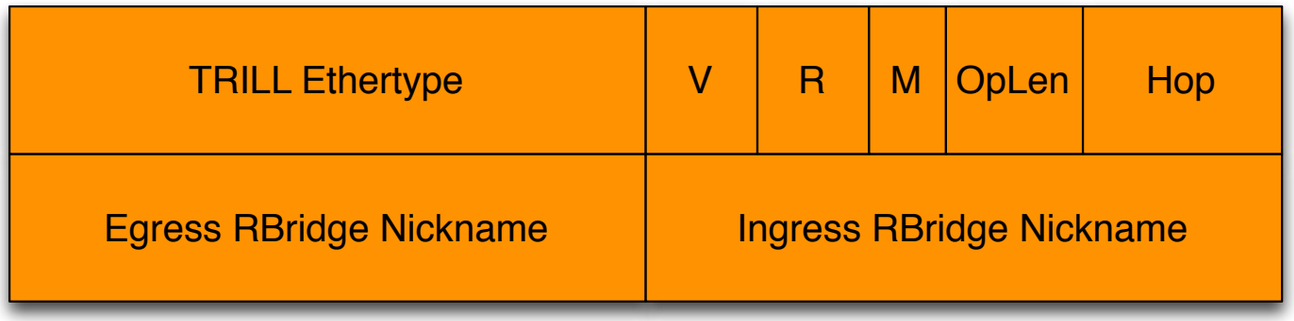
TRILL Frame Types

- TRILL Data Frames
  - Used for encapsulation of original frames
  - TRILL Ethertype
  - destination is unicast or All-RBridges multicast address
- TRILL Control Frames
  - Used for IS-IS between RBridges
  - L2-IS-IS Ethertype
  - Destination is All-IS-IS-RBridges multicast address

TRILL Frame Format

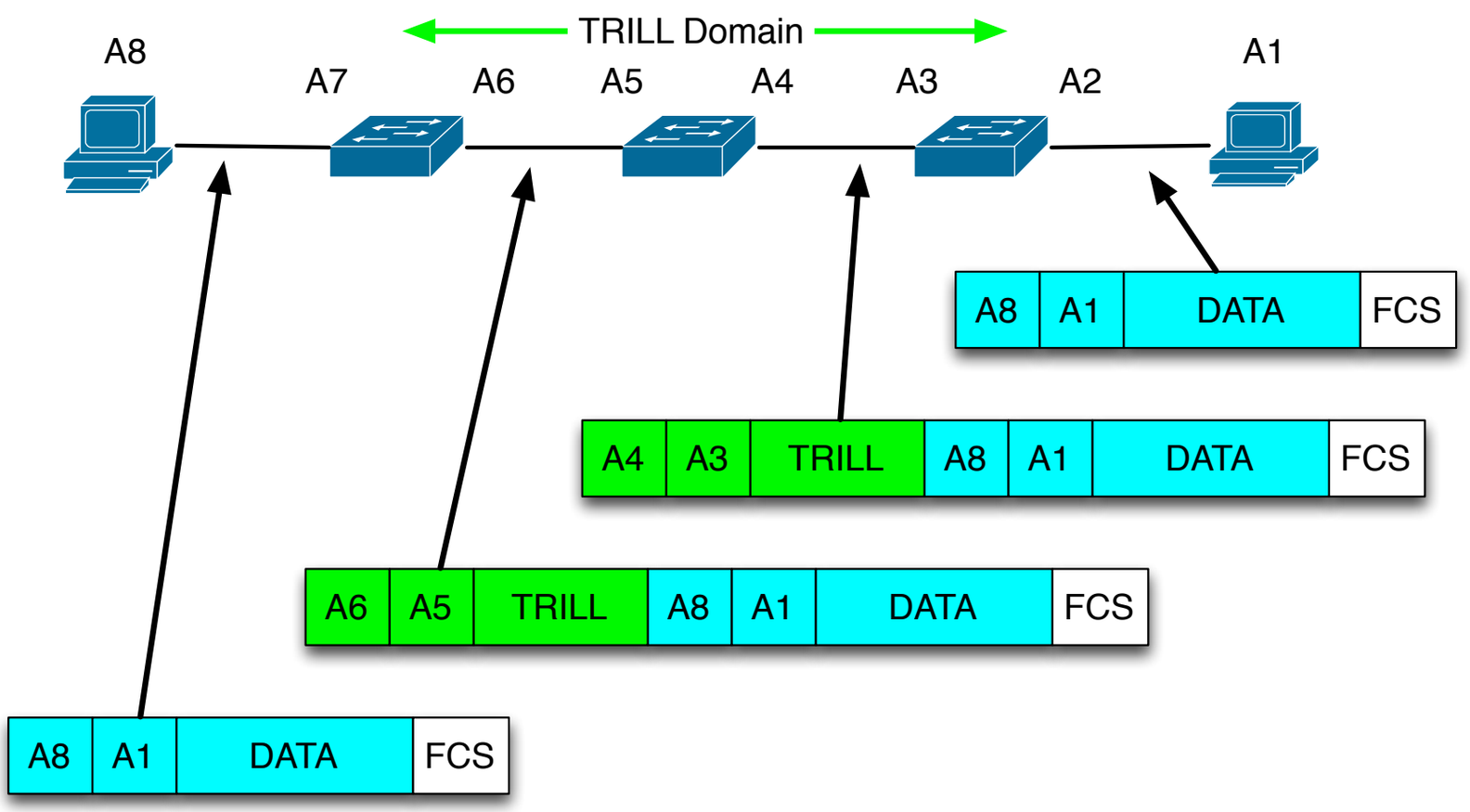
*) VLAN is optional

TRILL Header



Ethertype	16 bit
V: Version	2 bit
R: Reserved	2 bit
M: Multi-DST	1 bit
Option Length	5 bit
Hop	6 bit
Nicknames	16 bit

TRILL Forwarding



TRILL Unicast Forwarding

- IS-IS Hello Frames for topology discovery
- IS-IS reliable flooding to distribute global link-state database
- Unicast frames are forwarded hop by hop
- Hop count decremented at each hop

TRILL Multi-Destination

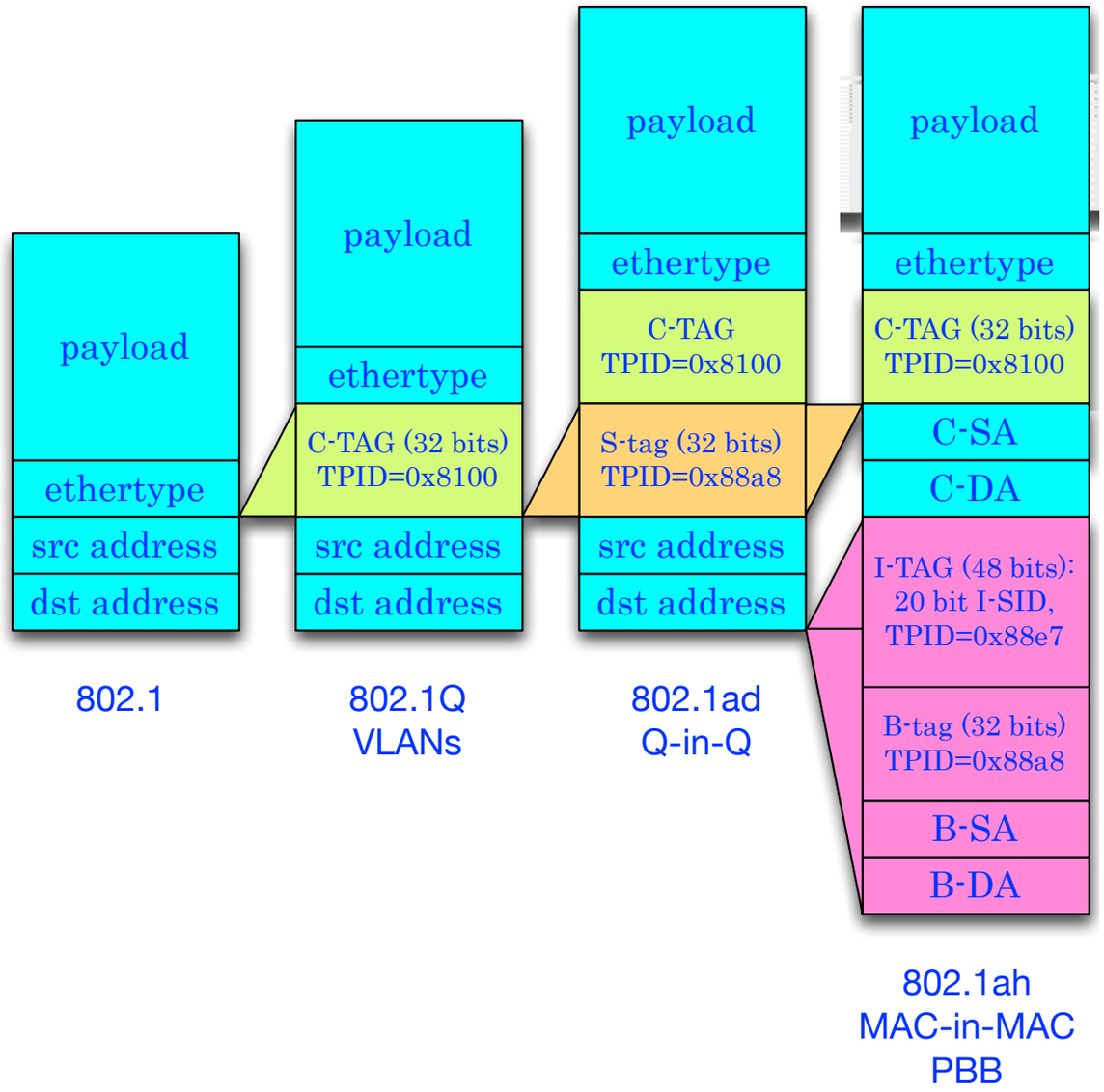
- Used for broadcast, multicast and unknown unicast
- Forwarded on a distribution tree selected by ingress RBridge
- Reverse Path Forwarding Check performed by each RBridge

IEEE 802.1aq

IEEE 802.1aq Basics

- Shortest Path Bridging
- Extension on either 802.1ad (Q-in-Q) or 802.1ah (Provider Backbone Bridges aka MAC-in-MAC)
- Uses IS-IS with minor TLV extensions
- Multiple shortest equal cost paths for both unicast and multicast (max 16)

Ethernet Evolution



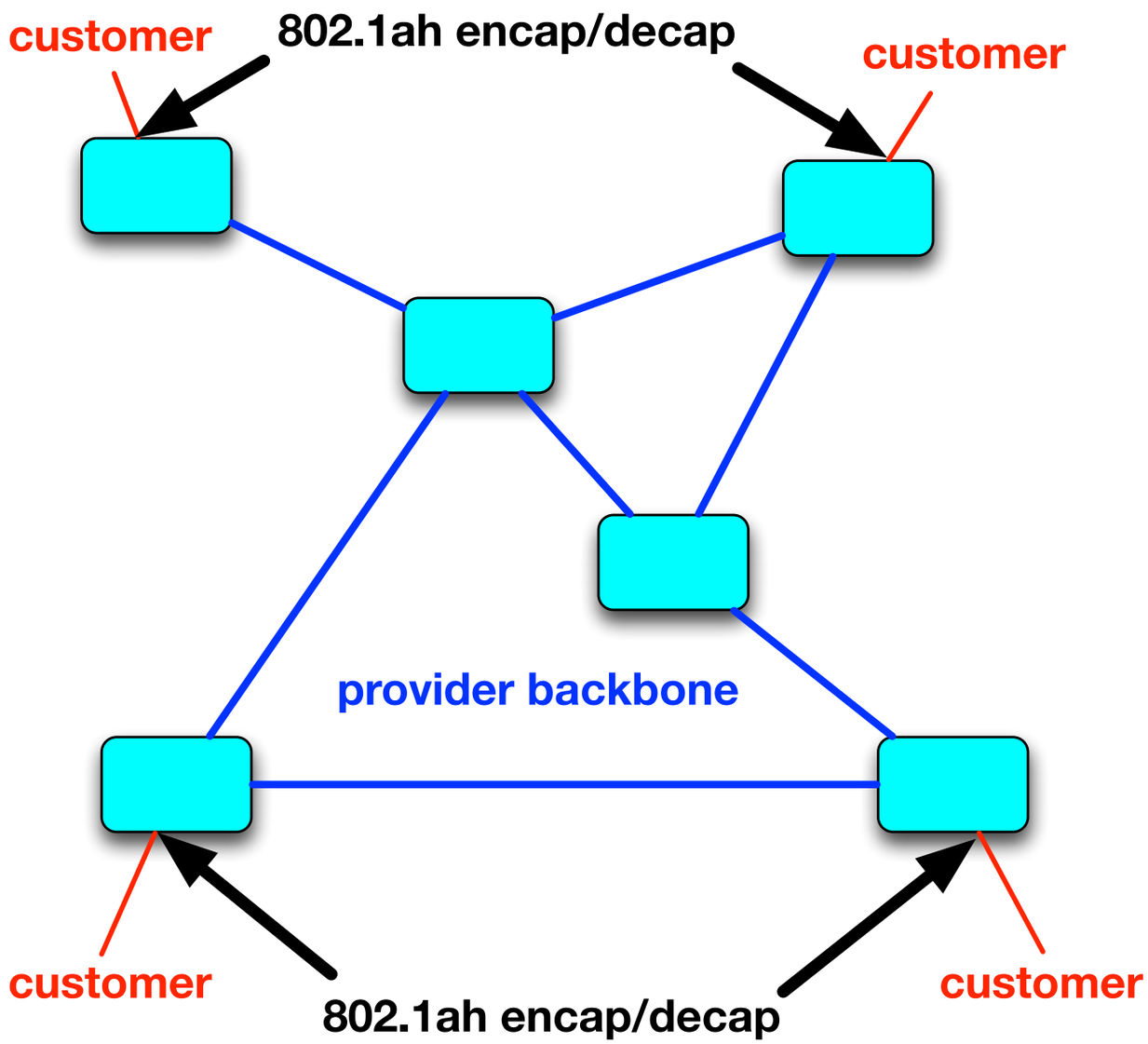
IEEE 802.1ad

- aka Q-in-Q
- additional VLAN tag
- Separation between customer and provider VLAN tag space
- Outer provider S-Tag
- Inner customer C-Tag
- Customer MAC addresses still learned at all provider switches
- Limited to 4K “services” (identified by S-Tag)

IEEE 802.1ah

- aka Provider Backbone Bridges
- aka MAC-in-MAC
- Encapsulate customer frame with provider Ethernet header
- Introduction of 20-bit I-SID
- 2^20 “services” (identified by I-SID)
- Customer MAC addresses only learned at ingress, not by switches within the provider network
- Outer MAC src/dst address (B-MAC) used to identify edge switches
- Switching within backbone based on B-MAC address of egress switch and backbone VLAN

802.1ah encap/decap



IEEE 802.1aq

- aka Shortest Path Bridging (SPB)
- SPBV (SPB-VID): extension to 802.1ad
- SPBM (SPB-MAC): extension to 802.1ah
- Most effort seems to go into SPBM
- B-MAC not learned but distributed via control plane
- Route to egress switch (B-MAC) via shortest path
- Forward and reverse path for unicast and multicast are symmetric
- IEEE 802.1ag OAM for Connectivity Fault Management

IEEE 802.1ag

- (this is ag, not aq)
- OAM protocol with normal Ethernet frames (Ethertype 0x8902)
- Three message types:
  - Loopback Message and Reply (LBM/LBR)
  - Link Trace Message and Reply (LTM/LTR)
  - Continuity Check Messages (CCM)

Loopback

- Loopback Message and Reply (LBM/LBR)
  - ping to MAC address
  - similar to IP ping
  - unicast Message, unicast Reply

Trace

- Link Trace Message and Reply (LTM/LTR)
  - trace to MAC address; replies from MAC addresses in path
  - similar to IP traceroute
  - multicast Message, unicast replies
  - uses TTL which is decremented by each switch in the path

Continuity Checks

- Continuity Check Messages (CCM)
  - periodic hello messages
  - used to detect link failures
  - sent as multicast
  - no replies, just processing of received CCM frames

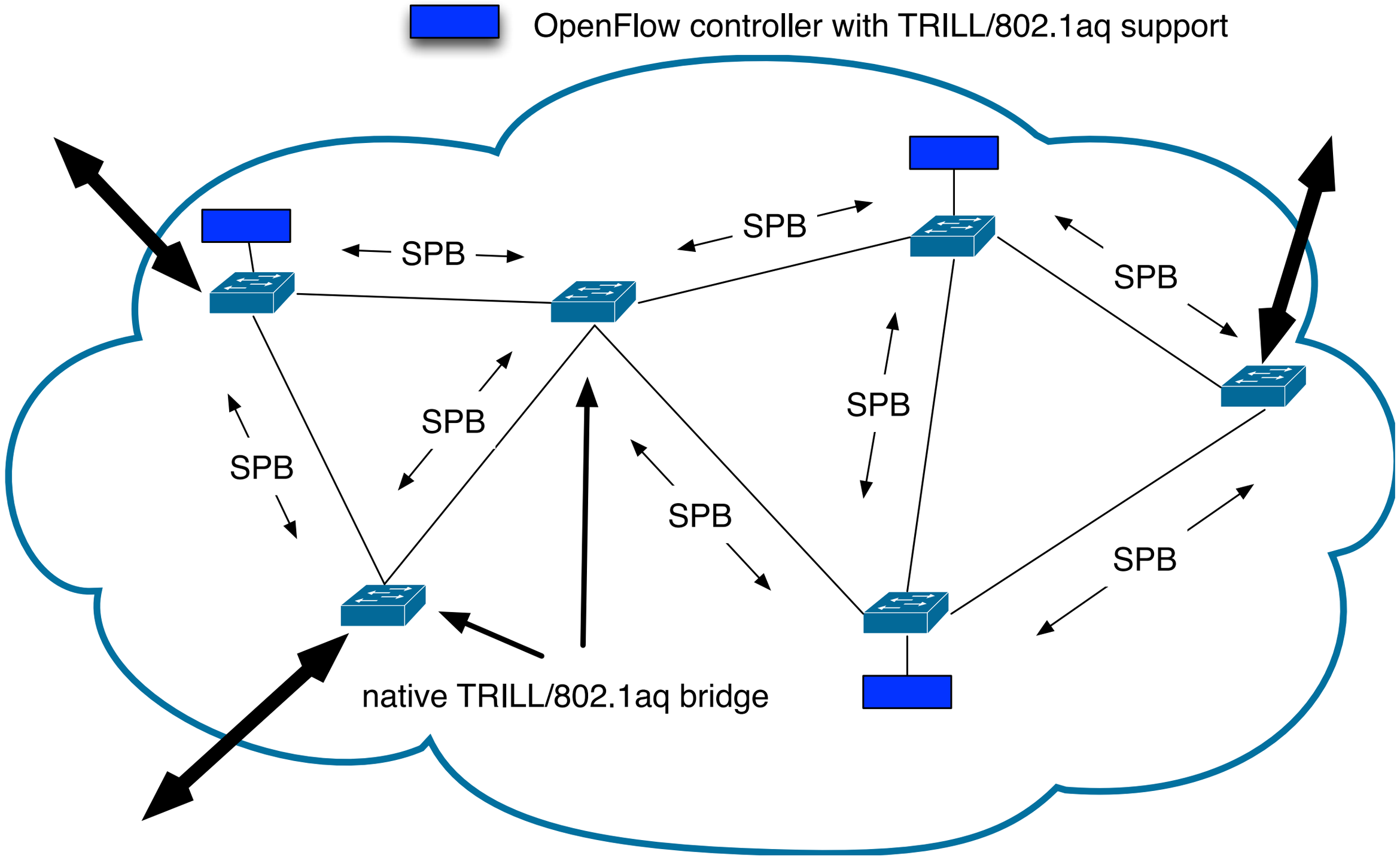
TRILL vs 802.1aq

(my view)

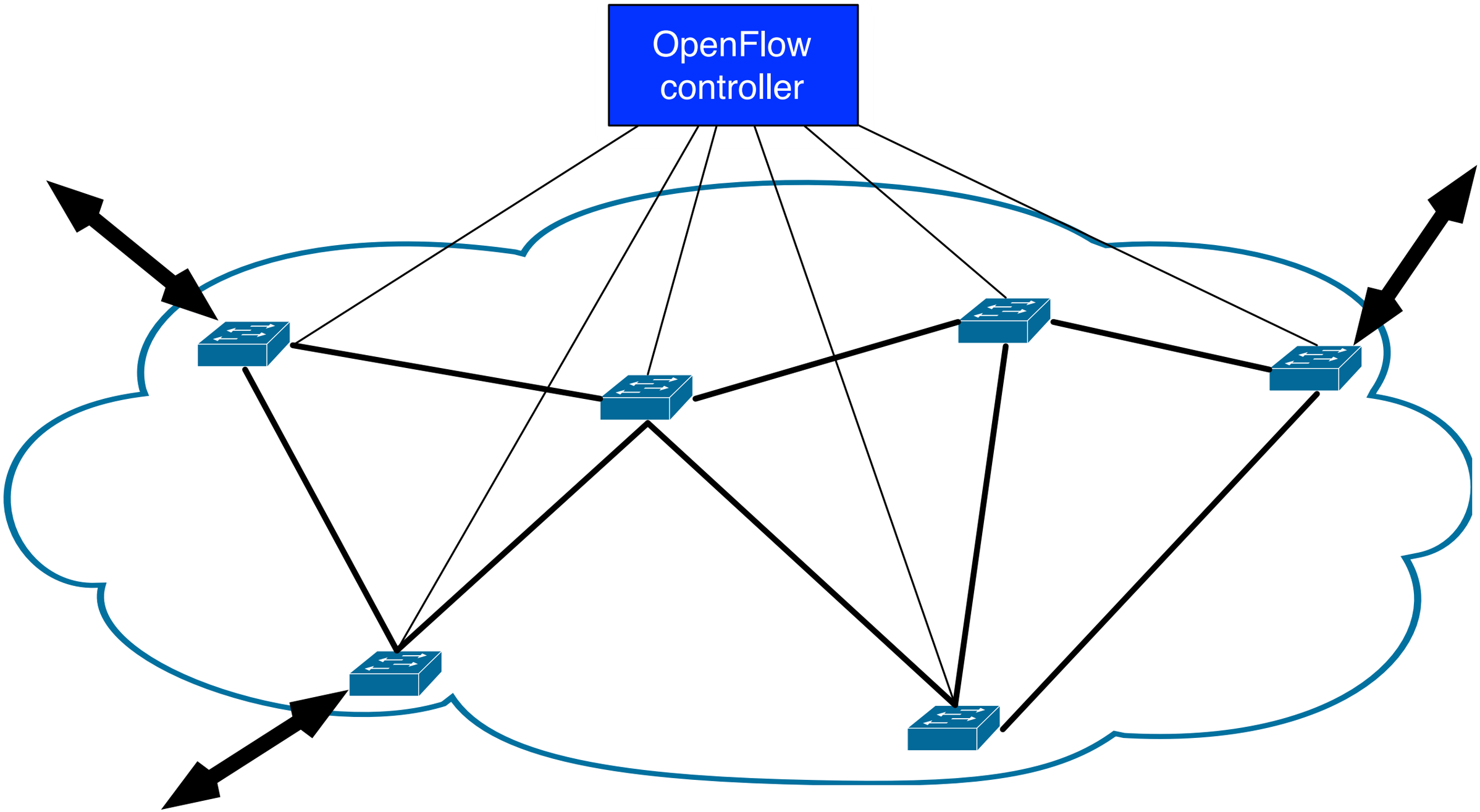
- 802.1aq uses normal Ethernet header (TRILL uses MAC src/dst + TRILL header)
  - 802.1aq just forwards
  - TRILL need to adjust Hop Count and outer MAC addresses (rewriting each frame)
- 802.1aq uses existing OAM (802.1ag); TRILL TBD
- 802.1aq sends multicast and broadcast via shortest path (TRILL does not)
- TRILL uses TTL as anti-loop protection (802.1aq RPF check more vulnerable)

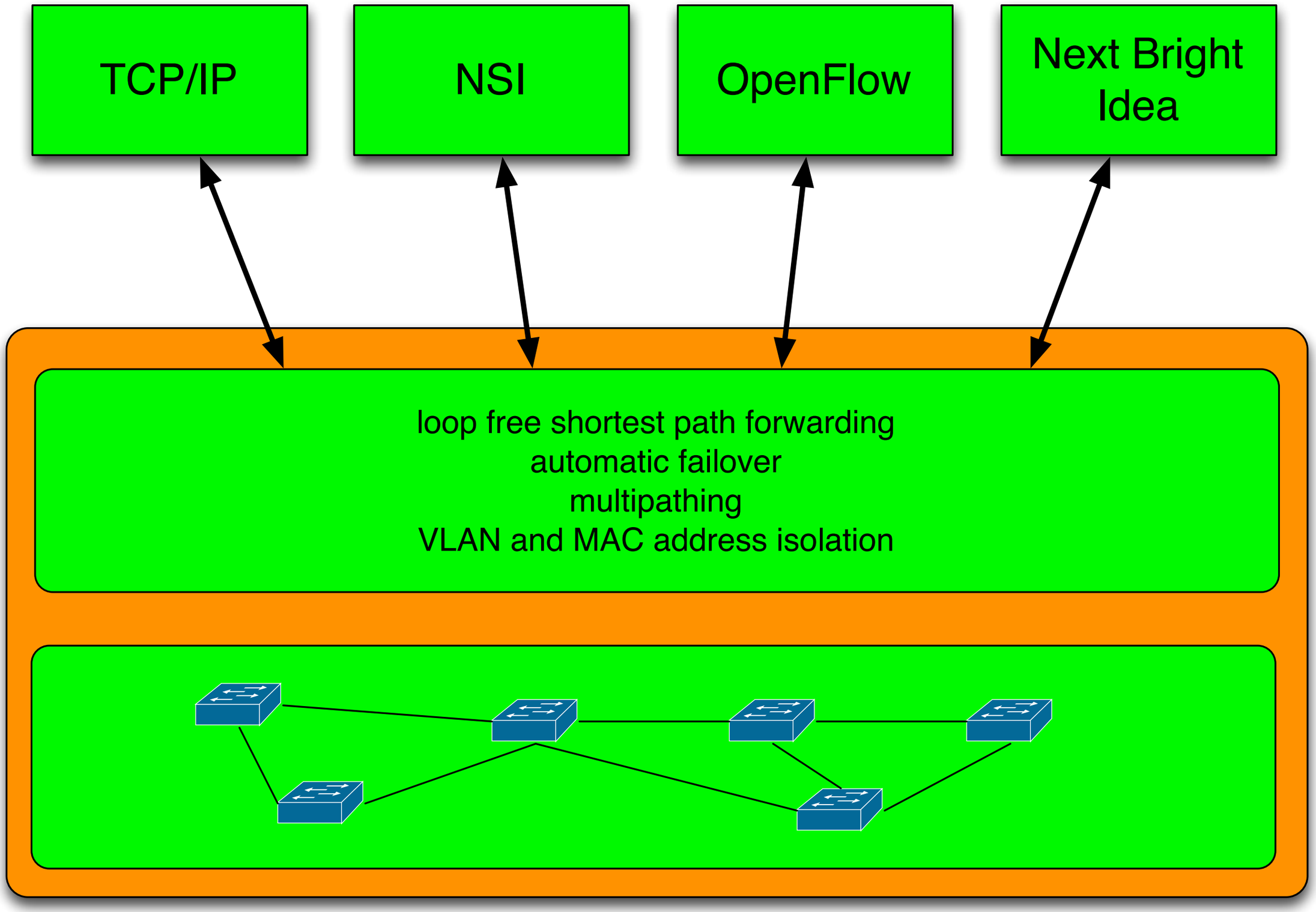
How to deploy SPB?

- As L2 clouds with SPB functionality
  - In switches running TRILL or 802.1aq protocol on the dataplane?
  - In an OpenFlow L2 cloud via OpenFlow controller?
- As part of the middleware?


OpenFlow controller with TRILL/802.1aq support
SPB
SPB
SPB
SPB
SPB
SPB
SPB
SPB
native TRILL/802.1aq bridge

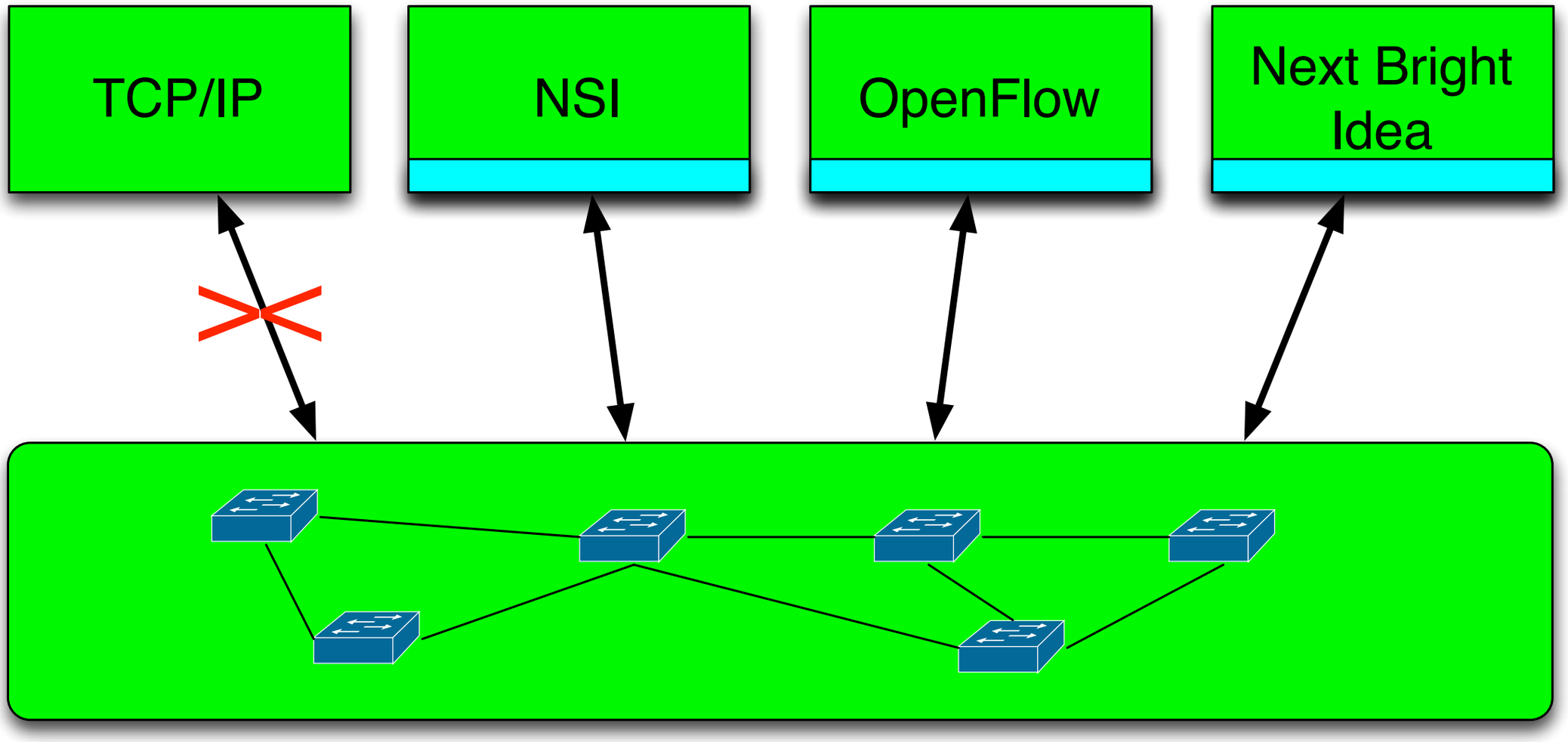
SPB intelligence in controller
switches send topology and reachability info to controller
controller pushes forwarding entries to switches




TCP/IP
NSI
OpenFlow
Next Bright Idea
loop free shortest path forwarding
automatic failover
multipathing
VLAN and MAC address isolation

SPB functionality replicated
in each "middleware"

loop-free shortest path forwarding
automatic failover
multipathing
VLAN and MAC address isolation



Discussion