All-Path Bridging Update (II)

Guillermo Ibanez (UAH, Madrid, Spain)
Jun Tanaka (Fujitsu Labs. Ld.)
Vinod Kumar (Tejas Networks)

IEEE Plenary interim meeting Munich 16-19 January 2012

Contents

- All Path specific features and value
- Issues raised in Atlanta meeting
  - includes the recently found direct compatibility with multiple isolated standard bridges (e.g. core switches in data centers)
- Additional results on path diversity and load distribution
- All Path protocol variants
- All Path Multicast optimization
- Conclusion
  - All Path initiates an additional and innovative path for the evolution of bridges

All Path specific features

- Is a generic forwarding mechanism to set up low latency paths or trees (multicast)
- All Path concept opens a new branch of evolution for bridges
  - Parallel to link state routing approaches
- SRP (AVB) belongs to the same branch:
    - SRP uses control frames, with additional QoS info
    - All Path instead reuses data frames to find paths
- With distinctive features:
  - Resiliency, low latency, load distribution

Issues raised in Atlanta meeting

1. Path oscillation/suboptimal path after recovery/congestion (1a and 1b)
2. Does All Path scope overlaps SPBV?
3. Cut-through switching in All Path
4. Multi-line chassis implementation challenge
5. Does All Path work with shared media ?
6. Path recovery
7. QoS . Prioritize packets?

1a Path Oscillation / Suboptimal Path after recovery

Question:
After a fast link recovers, path is no longer optimal, new path should be restored.

Answer:

- This would make sense in a protocol that sets paths/trees between bridges, not between hosts, like All Path.
- Hosts set up new paths on demand. If path aging is configured aggresively, old paths will expire fast and new paths will start using the recovered and light loaded fast link.
  - Assuming an hypotethical All Path between Bridges protocol (instead of the current All Path between hosts proposal, paths could be maintained on an average latency basis, but this is out of scope as it would overlap with SPBV and specific features like load distribution disappear.

1b Temporal congestion/suboptimal path

Question: Temporary congestion may result in long term sub-optimal path usage. As long as another flow is active, the new flow from/to the same hosts shall follow the sub-optimal path.

Answer:

- Reducing aging (refresh) timer forces path renewal more often. Pauses in the flows will age the path and trigger the set up of new path.
- The low latency path with temporal congestion, once the congestion dissapears, will get a lot of new traffic flows once congestion dissapears, this load will be diverted from the suboptimal path, improving its relative latency.
- Possible protocol variant: **open new flows per protocol port to increase path diversity** (**added functionality at edge bridges for path set up). Switches would need to snoop port.**
- Reroute to previous path is neither needed nor advisable (oscillation)

2 Does All Path scope overlaps with SPBV?

Question: SPBV and TRILL also cover small size networks and are zero-config.

Answer: Partially. Not in the lowest range of bridges. Offers simple, new functionality (low latency, on-demand load distrib.) without encapsulation and does not require topology knowledge.

- All Path is just a basic forwarding mechanism
- To overcome MSTP complexities, without tagging or encapsulation.
  - VLANs use in SPBV core derives from the need to prevent loops in MSTP Shortest Path Bridging approaches.VLANs are not needed for that in All Path, can be used in the standard way to create separate virtual networks in the infrastructure
- All Path offers natively important features: **low latency and load distribution with zero configuration**), suitable for many small and medium size networks applications.
  - Most applications benefit from low latency and load distribution
  - It could be suitable for basic (best effort) audio video bridging (something like ***best effort low latency bridging****)*

2 Does All Path scope overlaps with SPBV (cont.)?

- All Path is a **departure point** for bridging protocols, with many additions possible: All Path is just a basic bridging protocol plus a built in loop prevention mechanism (close in concept to RPF). Additions on top:
  - Low latency multicast trees. Path diversity.
  - Per Class of Service latency handling
  - Compatible with any tagging: plain VLANs, others.
  - Path Proxying and ARP Proxying on bridges
  - Flow based forwarding (SA-DA), port based
- Two basic flavours: Any packet SA learning (Tanaka) broadcast (ARP,ND) only (Ibanez)

3 Cut-through switching in All Path

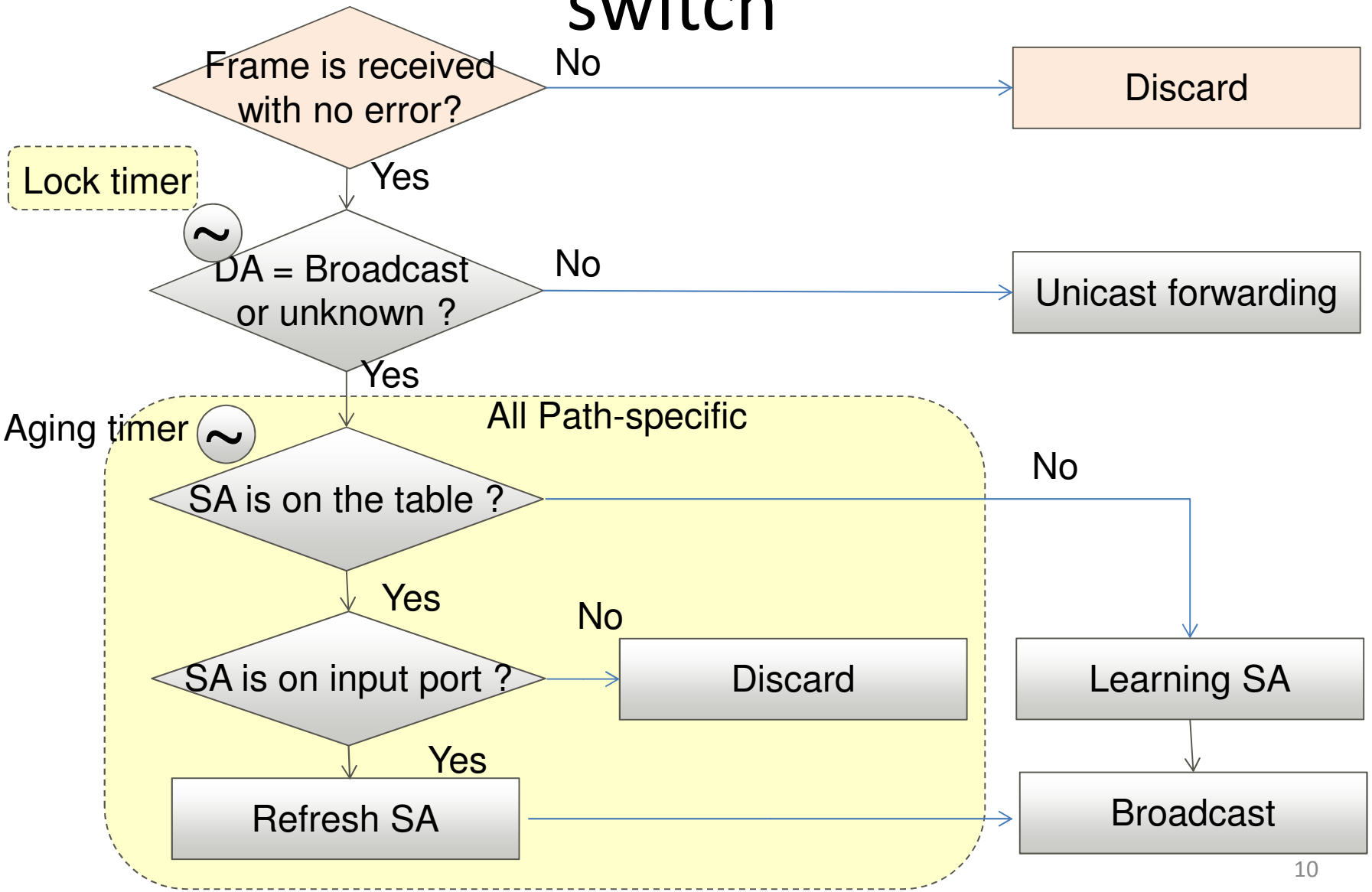
Question:

Is cut-through switching possible in All Path?

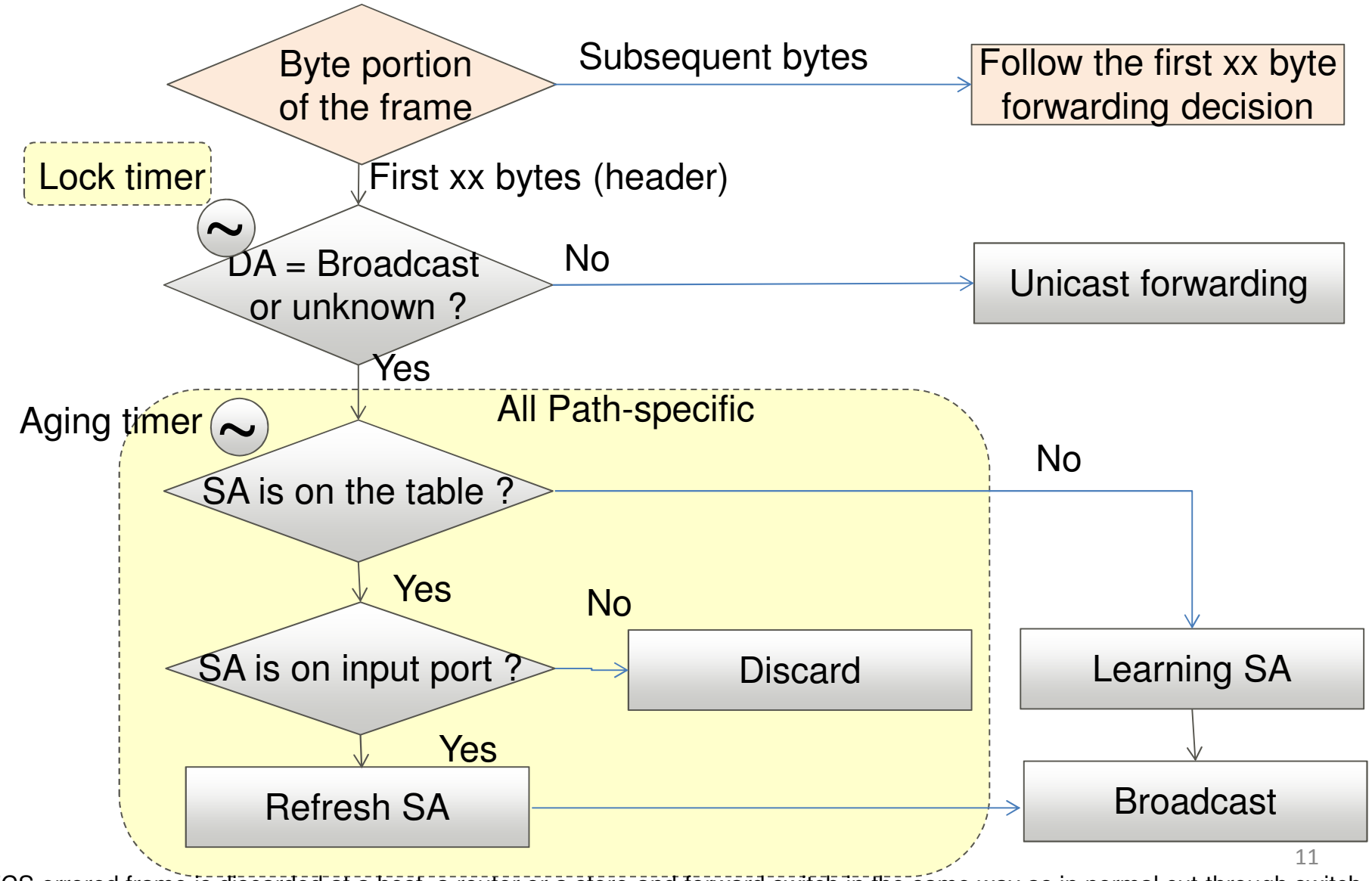
Answer: Yes

- *Broadcast frames: SA association to input port must be checked prior to forwarding via output ports*
- *Unicast frames do not strictly require this check,*
  - *because there is no problem in forwarding unicast frames received via ports not associated to SA, as long as DA is in the table (no danger of loops), and as long as there is no learning from these (any) unicast frames*

3. All-Path with Store-and-forward switch



3. All-Path with Cut-through switch



FCS errored frame is discarded at a host, a router or a store-and-forward switch in the same way as in normal cut-through switch.

4 Multi-line chassis implementation challenge

Question:

If we are going to implement this on multi-line card switch (chassis type) we need to synchronize each line card table at wire-speed. It might be unrealistic.

Answer:

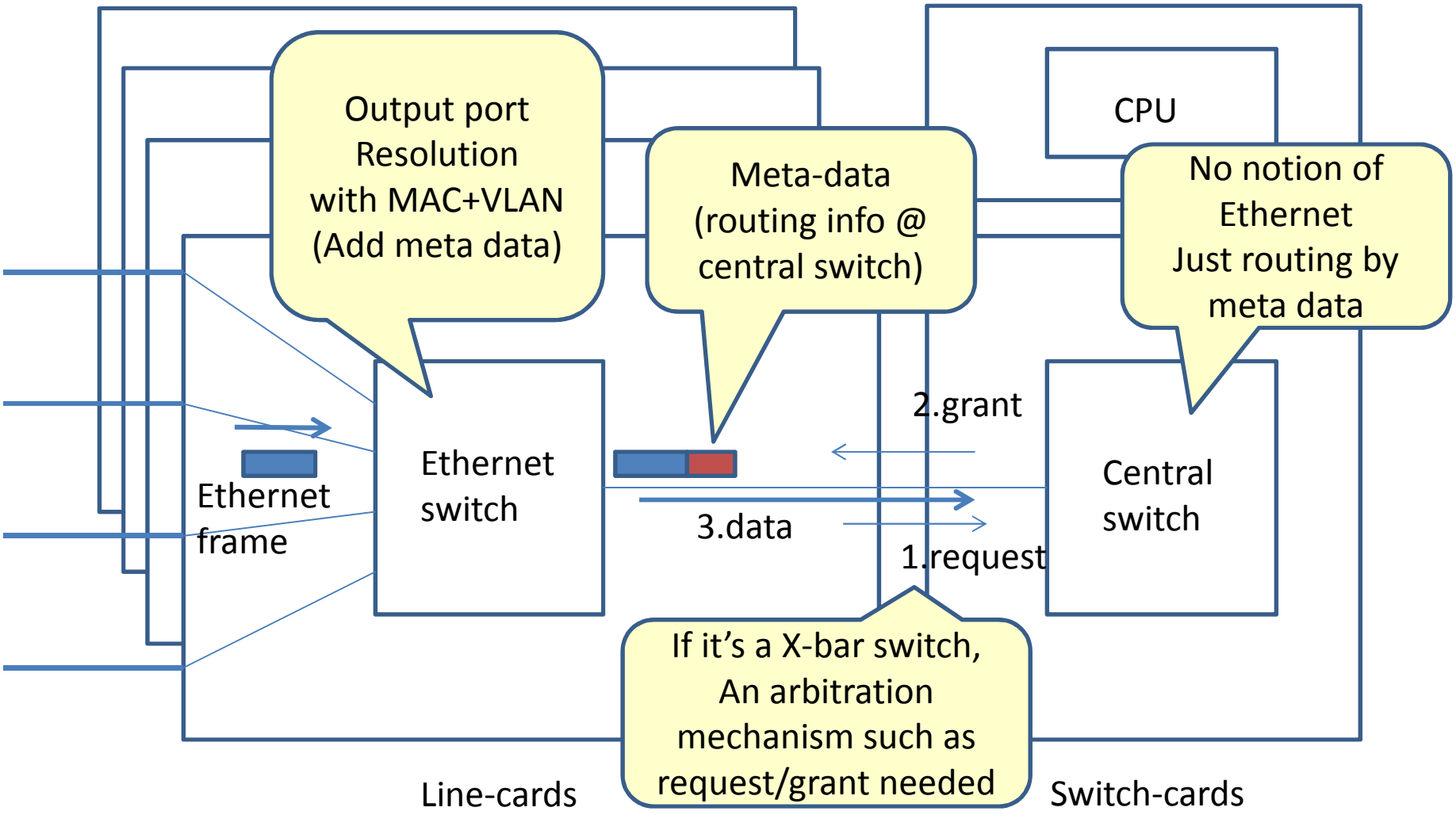
-This is a challenging hardware design that perhaps requires new approaches for address filtering implementations. A key point is arbitration to write into table. There is room for new designs.

-Use time stamping in line cards. See next slides.

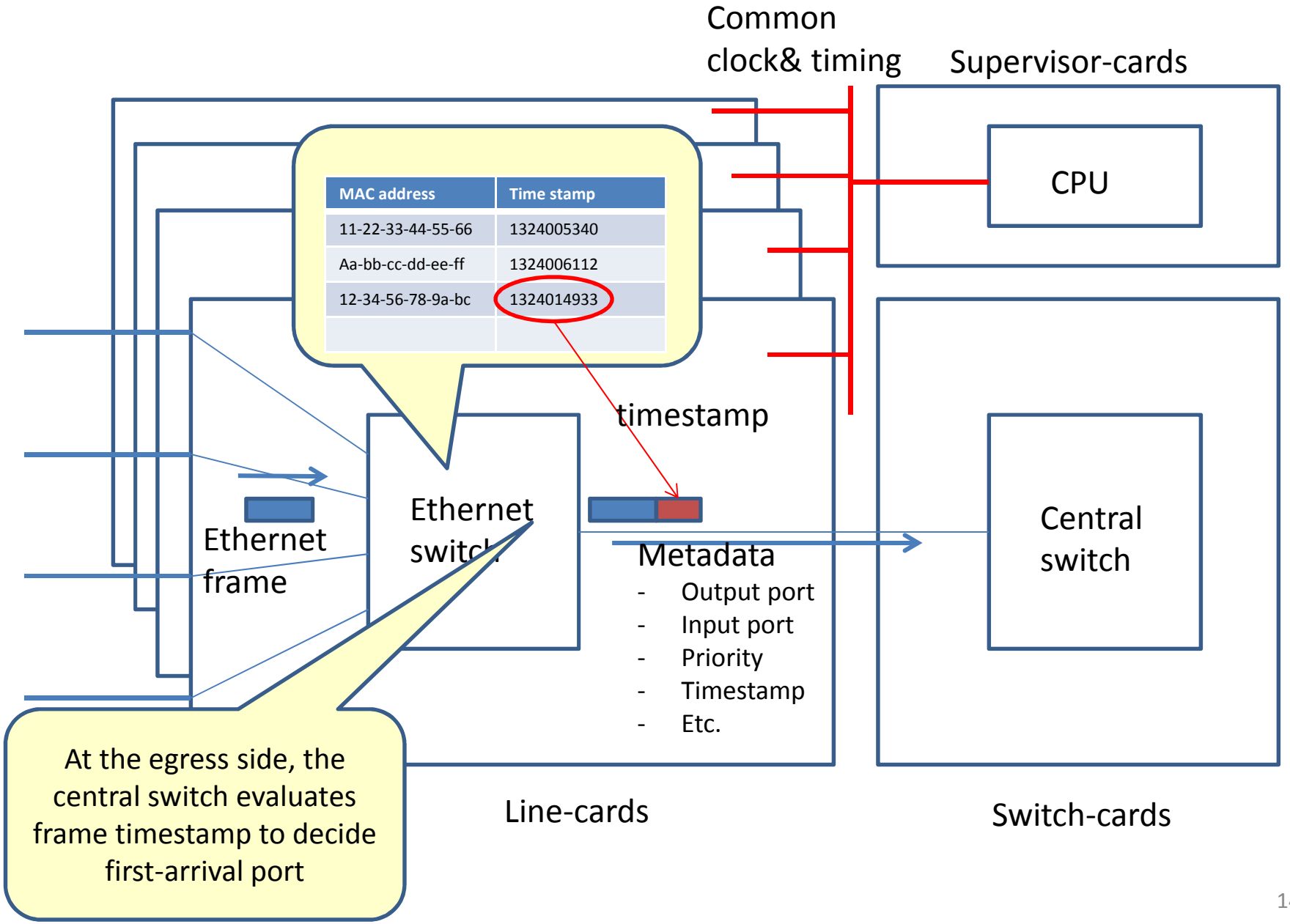
-Or use a stacking approach (e.g. 24*1 Gb + 2* 10 Gb uplinks), with integrated management .

4 Multiple Line Card Chassis Switch Implementation.

Ingress data flow

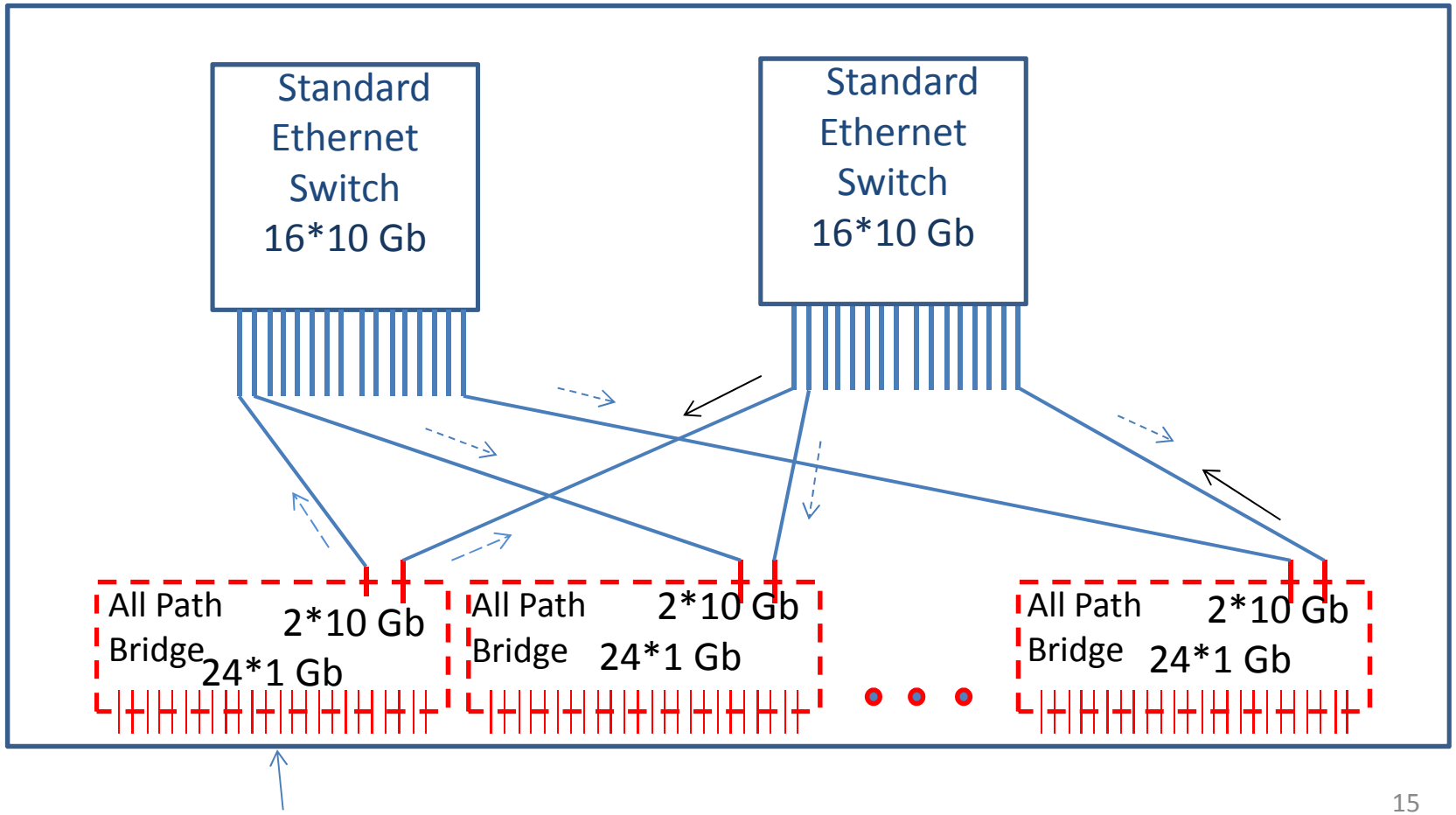


4 Multiple Line Card Chassis Switch Implementation.



4. All Path Bridges (APB) can operate with Standard Core Ethernet Switches!

No loops if only one core switch (isolated) per switching plane!
Fastest path is selected



5 Shared Media in All Path

Question:

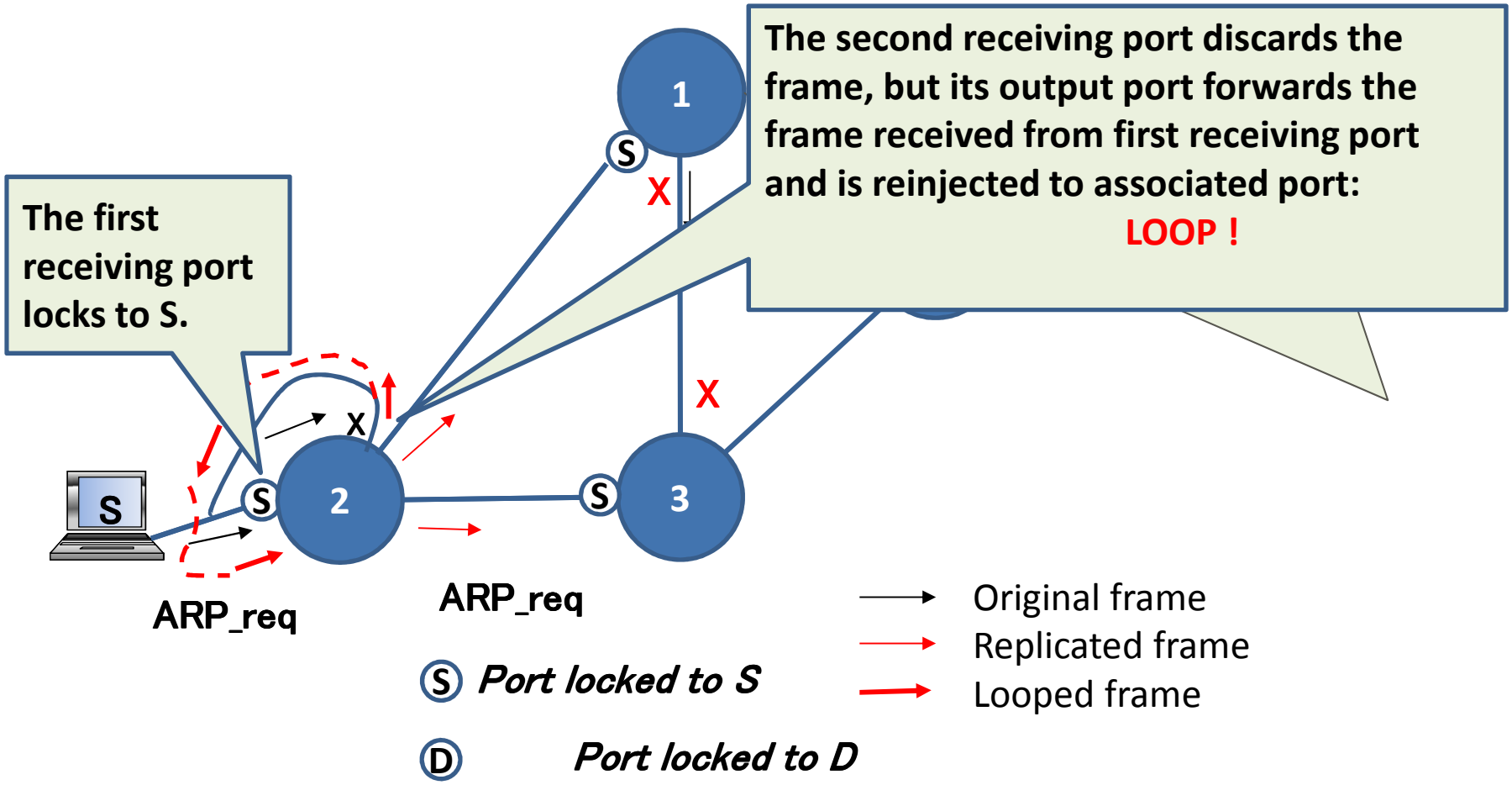
Does All Path work with shared media?

Answer: Basically NOT. Slides illustrate some loop situations

There are important restrictions:

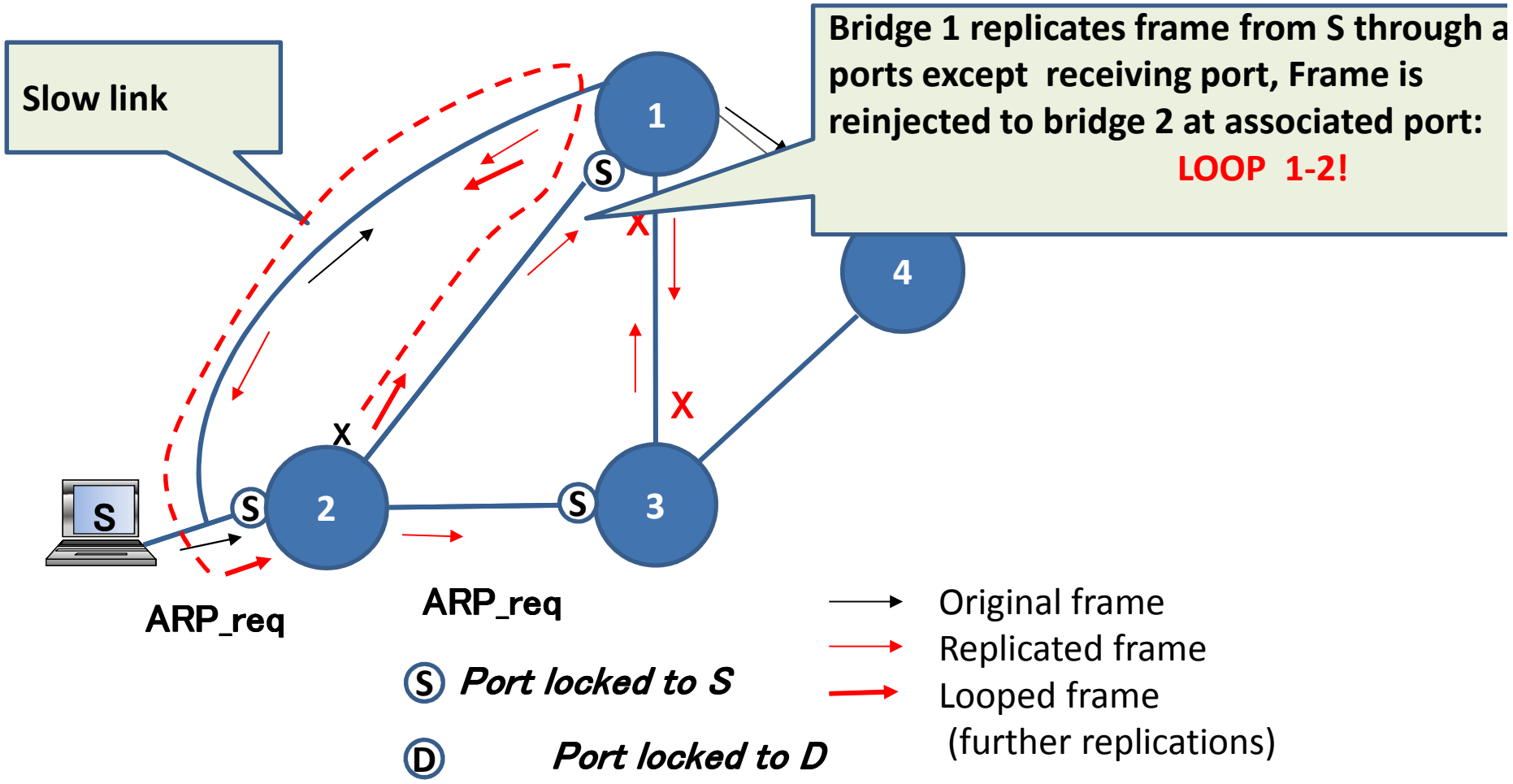
- Hosts may share a common link to a bridge but not to two bridges
- Shared links between multiple bridges and host(s) may create loops when they are slower than alternate paths by frame reinjection. See next slides.
- Protocol is intrinsically loop free using point to point links.
  - Looping ports of the same bridge does not create loops
- The subject has commonalities with interworking of 802.1D and All Path bridges: link aggregation, spanning tree protocol in connected 802.1D subnetworks

Shared media in All Path:
Loop condition with shared link (I, same bridge)

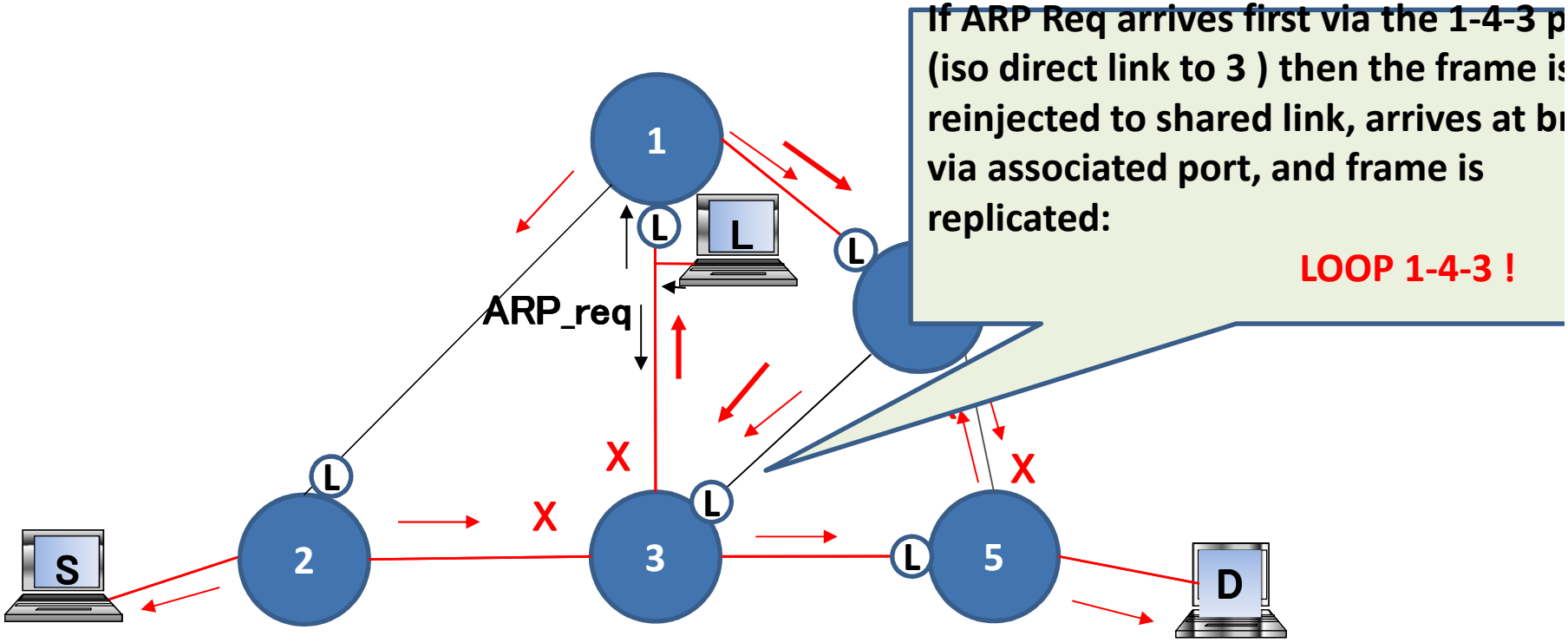


Shared media in All Path:

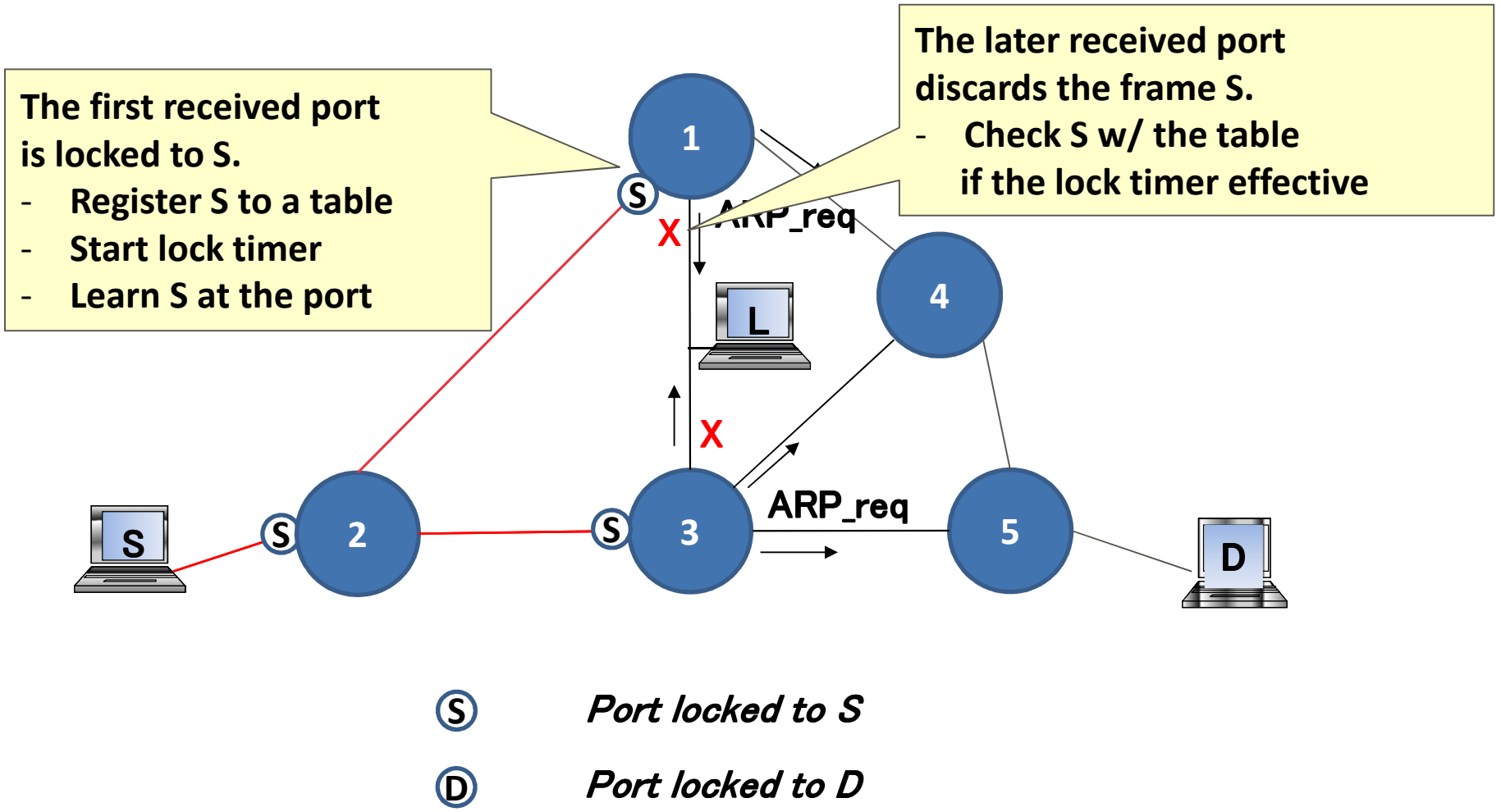
Loop condition with shared link (II, different bridge)



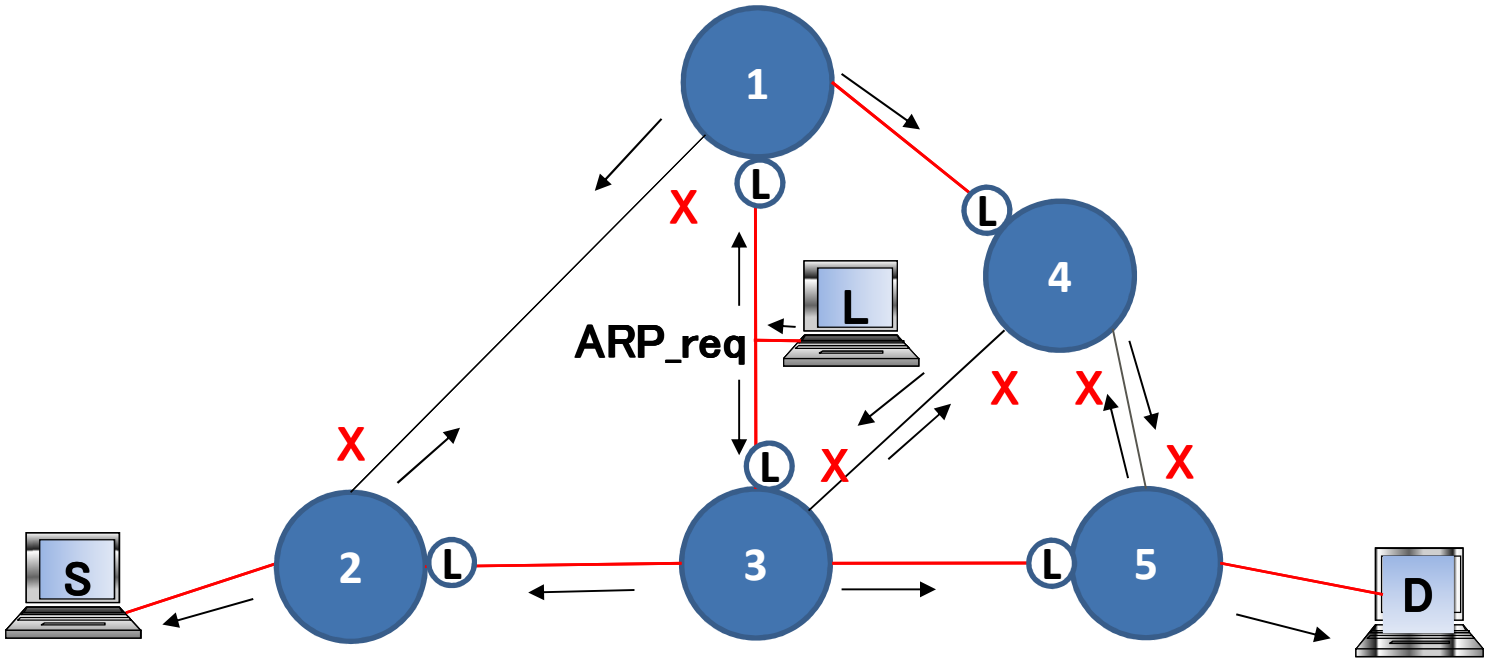
Shared Media in All-Path: broadcast from L may create loop if long delay to 3



Shared media in All Path: duplicate ARPs at host sharing link



Shared Media in All-Path: broadcast from L does not create loop (normally)



L — *Port locked to L (frames from L only accepted at this port)*

⟶ ARP_req

6a Path recovery

Question: Path Recovery resiliency to loss of flush packet

Answer:

- Total flush packet may be simpler to process and more reliable (can be repeated)
- Other path recovery methods are resilient:
  - Loss of a Path Repair packet loss or ARP Request packet has no effect as long as there is at least one path operative.
    - Intrinsic resiliency of using trying available paths

Path recovery

- Several methods possible (UAH version)
- Most simplified is:
  - Automatic loop back of unknown unicast data frames to respective edge bridge: distributes processing of path recovery among them.
  - Upon reception of looped back frame at edge bridge, edge bridge generates Path request (or ARP Request). Path Reply/ARP Reply
  - Only active flows recover the path, and when needed (distributed effort)

6b Path Recovery (II)

Comments: Generating ARP in bridge on behalf of a terminal is not a preferable approach.
The bridge should not be aware of L3 protocol such as IP.

Answers:

- ARP Path protocol can be seen as an “ARP Snooping” protocol (for IPv4)
- Protocol can adapt to the corresponding L3 protocol (for IPv6, Neighbor Discovery packets used iso ARP packets)
- For L3 independence, a Path_Fail packet, addressed to the All_Path_bridges multicast address, containing the destination host address can be used.

- *It is a design option to make it L3 independent*

7 QoS

Question: You should take care of QoS.

-How do you handle ARP and data frame during congestion?

- Prioritize ARPs vs Data frames?

Answer:

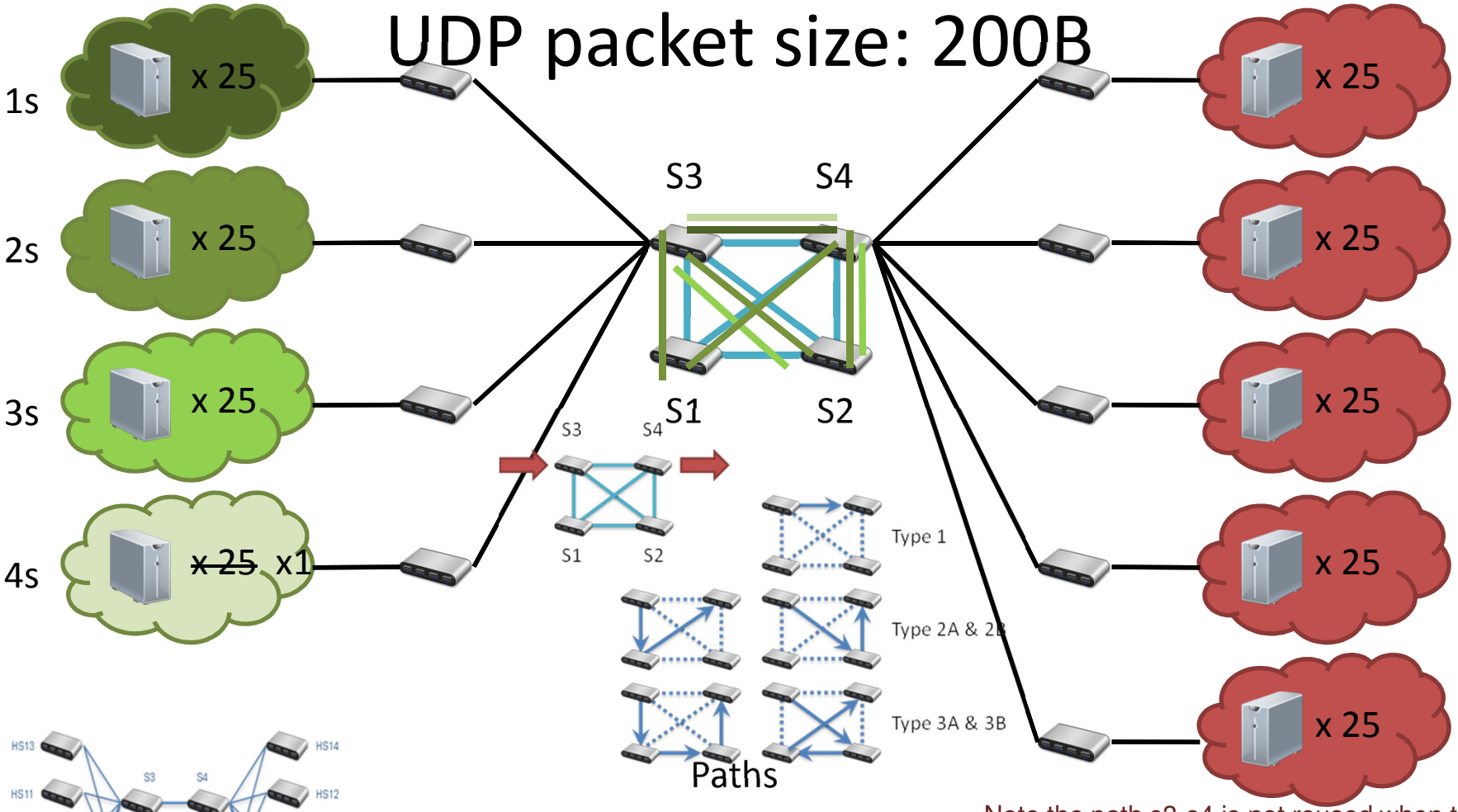
- Do not prioritize forwarding ARP vs data frames, if we want the path found to behave with **similar latency** for data frames.
- Need for some prioritization:
  - Path Repair frames should have priority (established paths) vs ARP (new paths)
  - Congested switches should not prioritize ARPs (accept new load) but even the opposite as an additional congestion control.
  - Future work: how to prioritize between ARPs from different flows.
    - Requires use of CoS bits or other priority assignment to flows

Contents

- All Path specific features and value
- Issues raised in Atlanta meeting
- Direct compatibility with core standard bridges
- Additional results on path diversity and load distribution (Flow model simulator and Omnet packet simulators)
- All Path Variants
- Multicast

Forcing Path Diversity (Omnet) among different latency paths

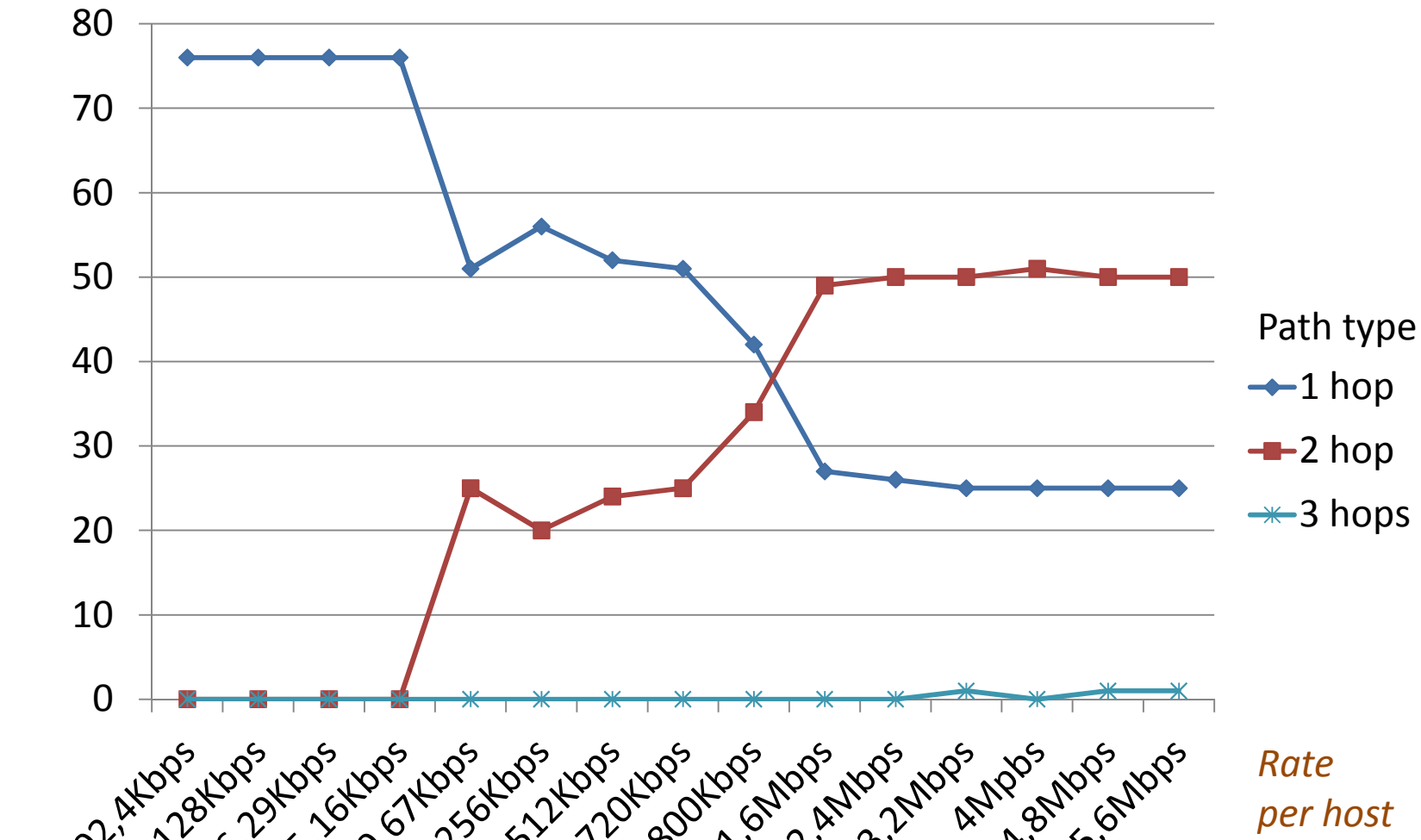
UDP packet size: 200B



Note the path s3-s4 is not reused when the 2nd group starts, but instead uses s3-s1-s4 and s3-s2-s4, similar with the 3rd group, the 4th reuses s3-s4 because it's again the fastest once s1 and s2 are loaded because of groups 2 and 3

Paths with increasing load

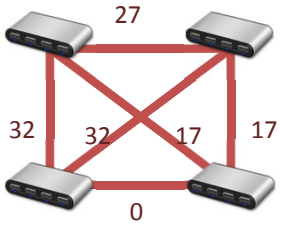
(traffic from left to right)



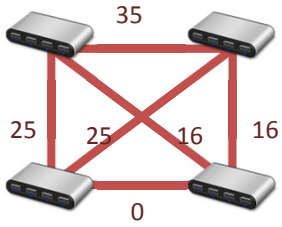
Path diversity vs propagation and processing delay (200B frames)

flows per link

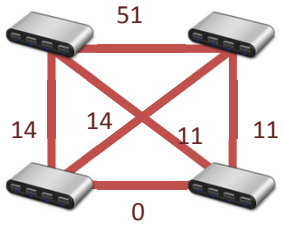
Queue size →100.000 Frame size → 200 bytes
Frame frequency→ 1ms Link speed → 100Mbps
Rate per host → 512Kbps Rate per group (x25) → 12,8Mbps



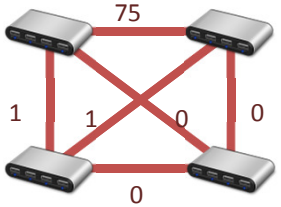
2us



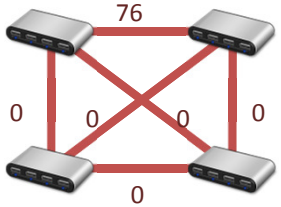
5us



10us



15us



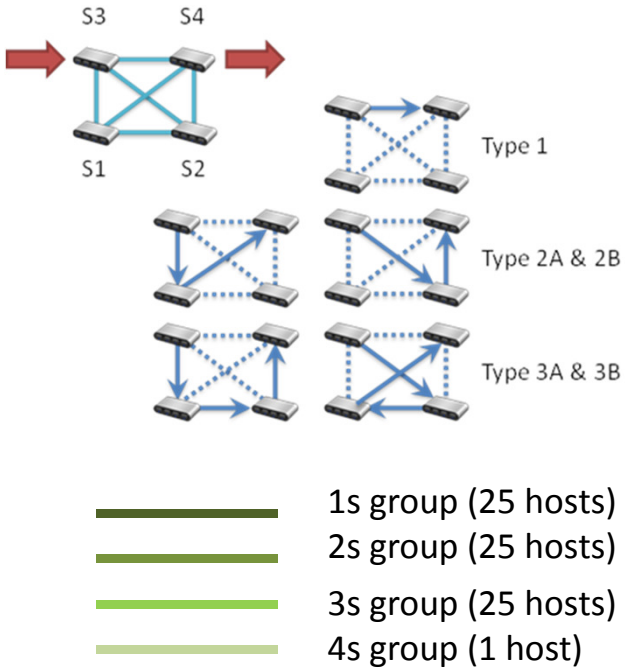
20us

Increasing processing delay at all switches reduces load distribution to alternate paths with higher latencies. The effect is similar for increasing link propagation delays.

Omnet simulation results, variable processing delay (200B)

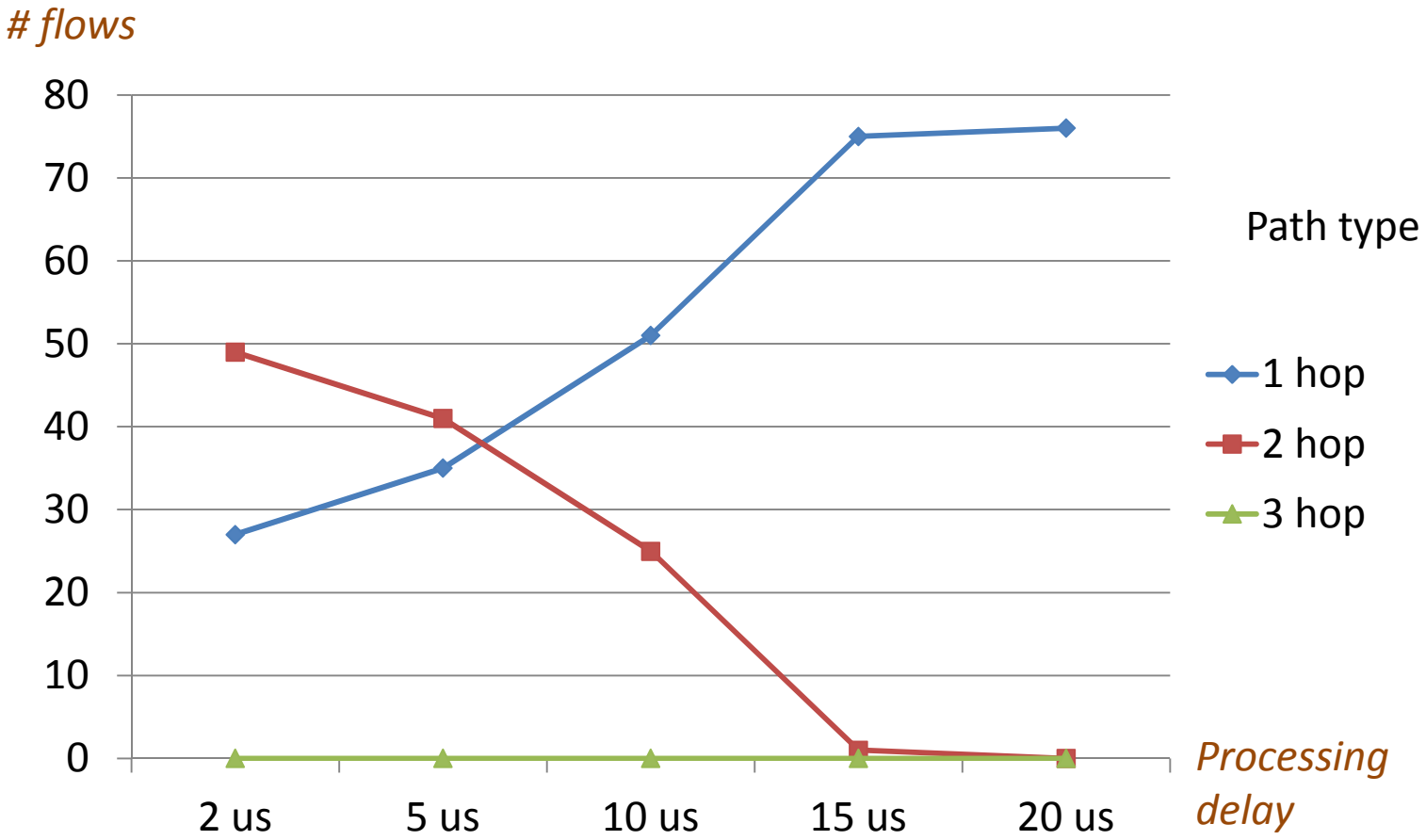
Queue size →100.000
Frame size → 200 bytes
Frame frequency→ 1ms
Link speed → 100Mbps
Rate per host → 512Kbps
Rate per group (x25) → 12,8Mbps

Delay	# flows per path				
	Type 1	Type 2A	Type 2B	Type 3A	Type 3B
200B → 1,6Mbps → 40Mbps (2us)	**25** + 1 + 1 = **27**	24 + 8 = **32**	17 = **17**	0	0
5us	**25** + 2 + 7 + 1 = **35**	23 + 2 = **25**	16 = **16**	0	0
10us	**25** + 25 + 1 = **51**	14 = **14**	11 = **11**	0	0
15us	**25** + 25 + 24 + 1 = **75**	1 = **1**	0	0	0
20us	**25** + 25 + 25 + 1 = **76**	0	0	0	0



Load distribution vs. processing delay

200 B (for traffic from left to right)



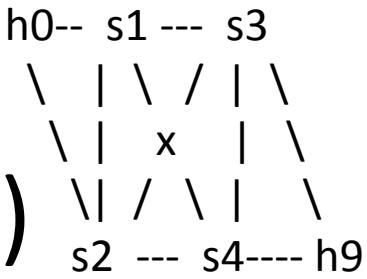
Load distribution with flow model simulator

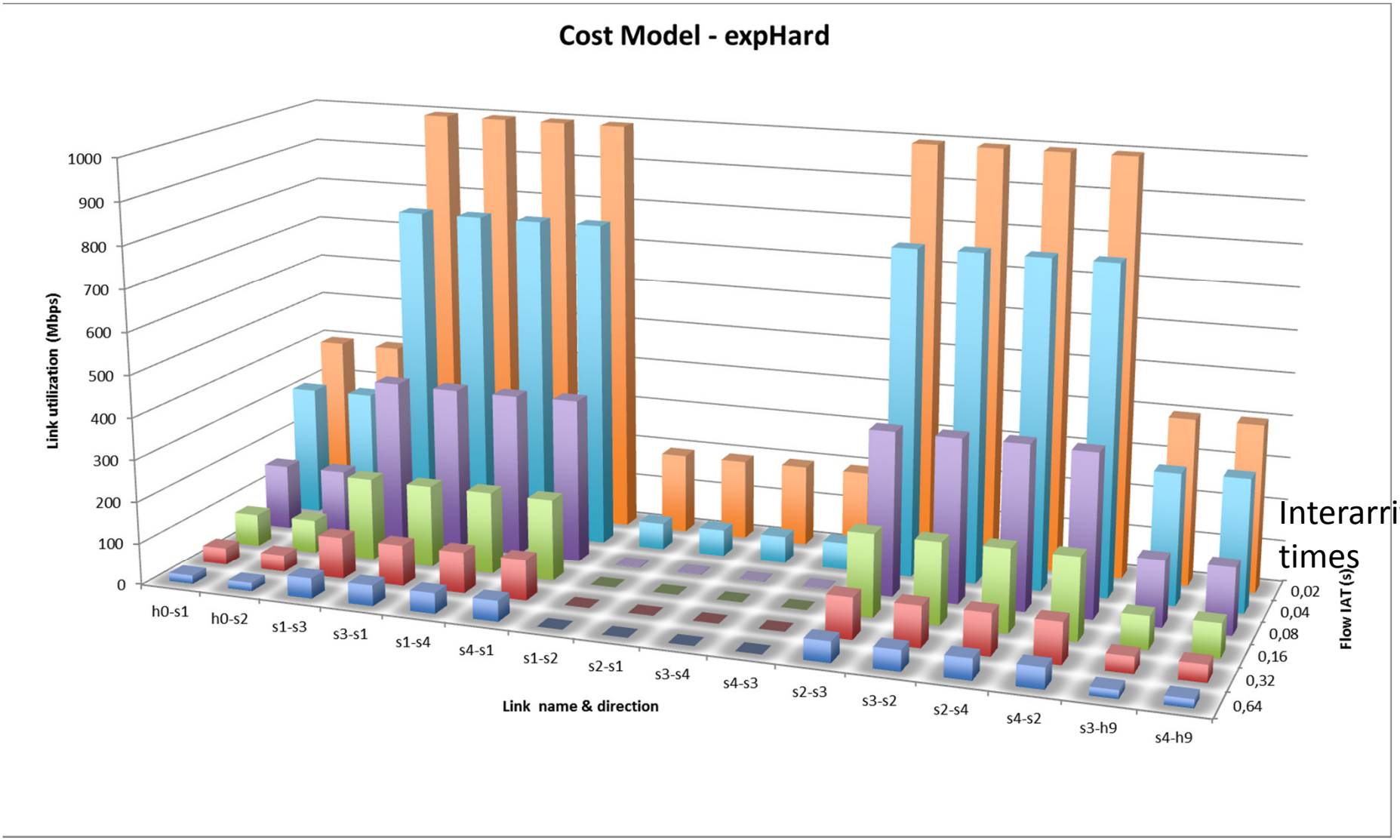
- Flow model based on:
  - [1] Ravi Prasad and Constantine Dovrolis, “Beyond the model of persistent TCP flows: open-loop...”
  - [2] Amund Kvalbein, Constantine Dovrolis and Chidambaram Muthu, “Multipath load-adaptive routing: putting the emphasis…”
- SIMPY discrete event simulator (Python).

Load distribution (Flow model simulator)

- Simulations with flow-model:
  - Recursive path selection based on lowest cost path available at that moment
  - Link cost depends of load (number of flows assigned per link) with cost randomization
  - Cost model :ExpHard: (10.000/link speed)/1-%load (similar to queue delay). Randomized over the range (0 to obtained value)
- Flow model simulation results show :
  - Uniform load distribution when alternate paths of closer cost exist
  - Path selection is determined by *differential* latency between paths, not on absolute latencies
    - Independence of absolute propagation delays
    - Low sensitivity to link cost models if number of flows high
  - Dependency on topology:
    - Excellent load distribution if alternate paths with similar latencies available
    - No distribution when paths have big latency differences
      - Could be tuned up assigning different priory queues at switches

Load distribution (all hosts sending Left-Right)

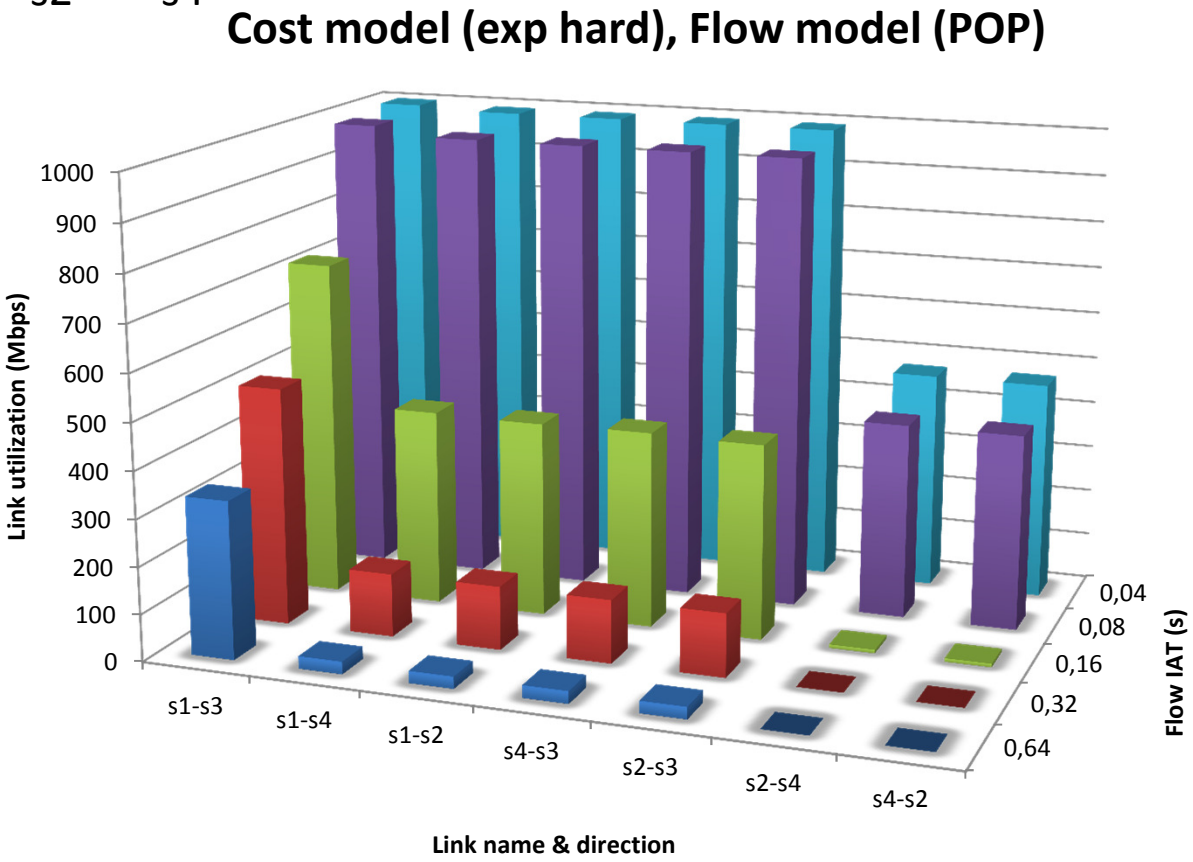




Interarrival times

Forcing path diversity

```
hosts --  s1 ---  s3 --
          | \   /  |
          |   x    |
          | /   \  |
          s2  ---  s4
```



All Path Protocol variants

- All Path between hosts: set paths between hosts
  - New variant with per-flow path establisment (SA&DA)
    - Path established per SA&DA pair
    - Increased path diversity, increased load balancing
    - Bigger tables but not as big as supposed (2-3 times)
- All Path between bridges: set trees from bridges
  - Same mechanism (first arrival port) for finding paths between hosts may be used **to set up trees between bridges** with multicast frames (slides shown in Volterra 2009)
    - Increased scalability, but reduced path diversity
    - Set_Tree messages
    - Interesting for **Multicast** scenarios
    - Other uses would overlap with SPBV.
    - Enforcing tree congruency complicates the protocol

All Path Trees construction

- Periodic broadcast of beacon frames from root bridges to keep alive tree paths
  - Tree path expires by link failure or timer expired
  - Stability of path is the priority: change only on path failure
  - Sophisticated mechanisms possible to decide changes on tree branches
- Multicast data frames can also be used to create and maintain trees
  - Simple mechanisms to prevent repeated broadcast over redundant links

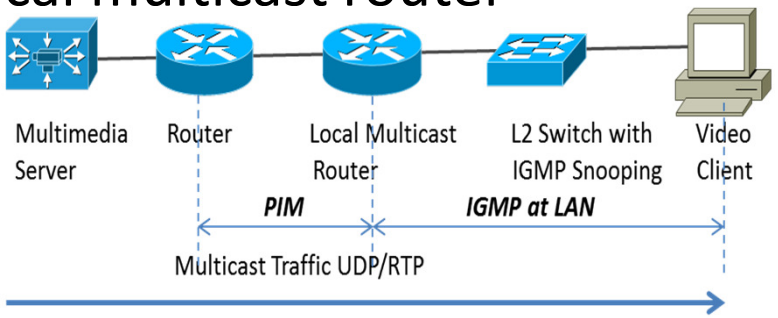
ALL PATH MULTICAST

All Path Multicast

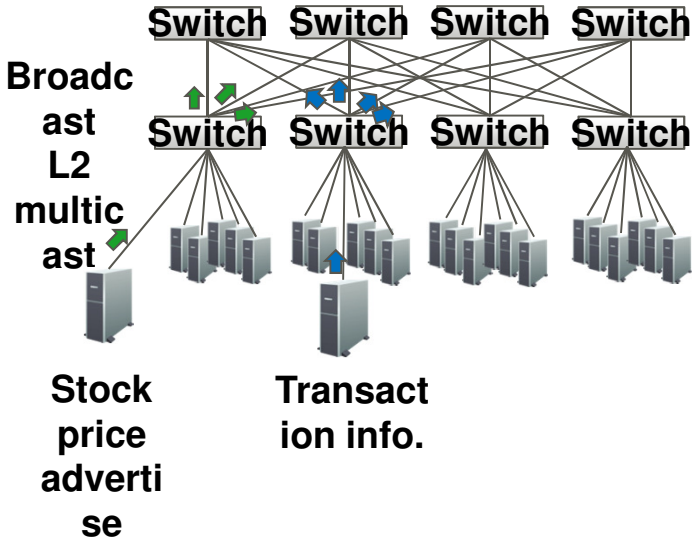
- All Path protocol is well adapted to multicast
  - Becauses it uses broadcast or multicast to find paths
  - Similar IGMP snooping as standard bridges
    - But possible on all active links
    - Potentially greater path diversity
  - Well suited for multiple source multicast scenarios
    - Builds naturally specific, low latency per group tree
    - Tree adapts dinamically to load when it is rebuilt
    - Tree diversity, adapted to links loads

All Path Multicast with IGMP snooping

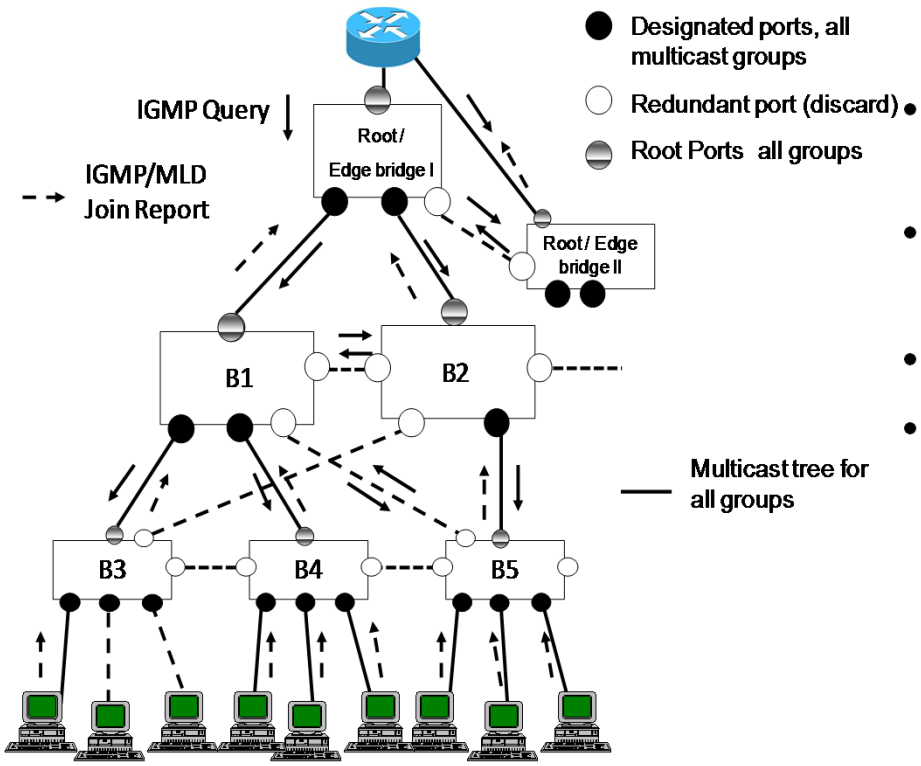
- Two scenarios:
  - One main multicast source: local multicast router



  - Multiple sources, financial cloud



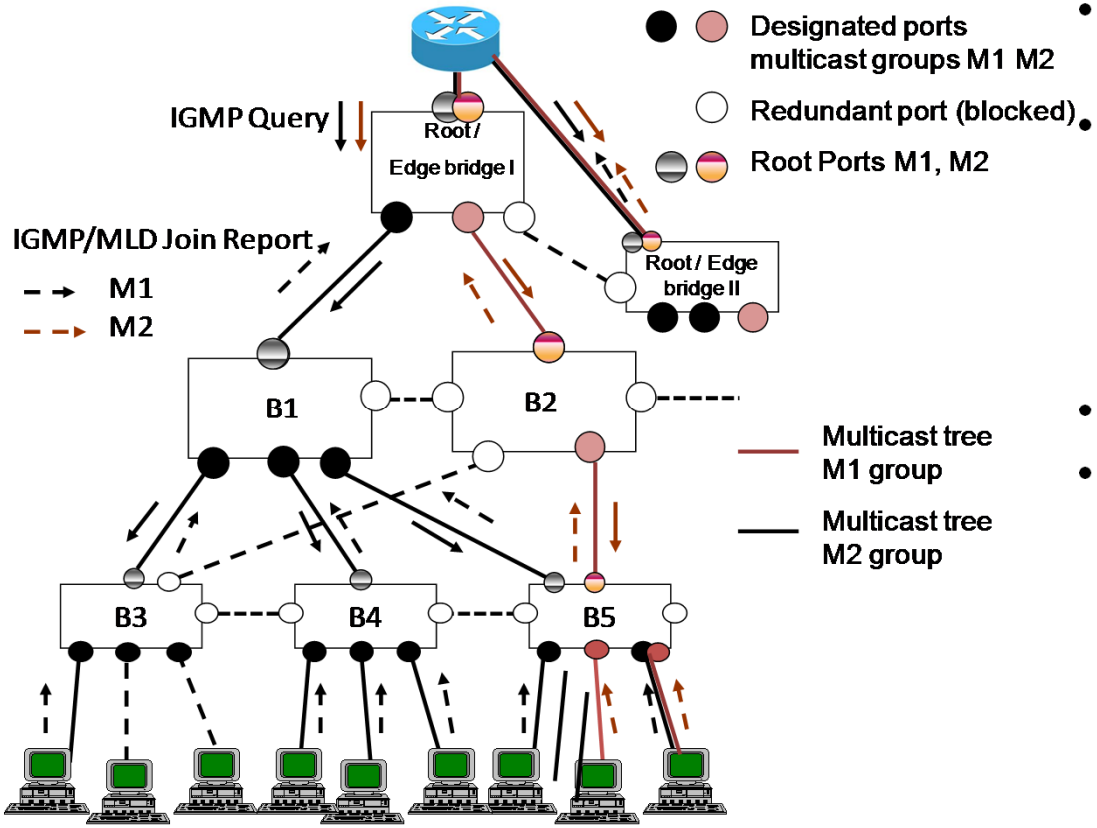
IGMP snooping, single tree



- Port roles are similar to RSTP:root, designated, redundant (alt./back up)
- IGMP Query/MLD sent by router or Querier switch
- Query is forwarded through All links
  - Discarded at redundant ports
- IGMP Query and Join packets are snooped
- Snooped Join confirms branch
- Unconfirmed tree branches are pruned accordingly

All Path single multipath tree with IGMP snooping

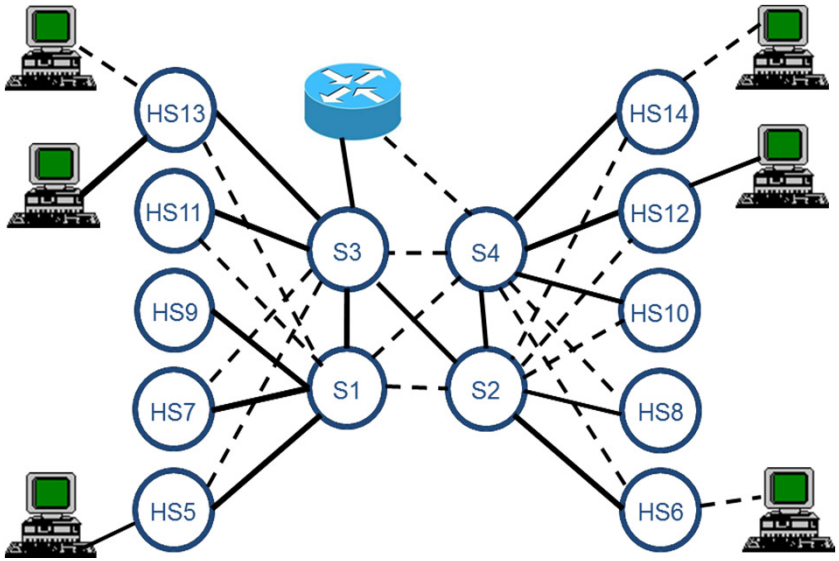
IGMP snooping, multiple tree



- One tree per set of multicast groups
- Tree branches diversity is proactively enhanced
  - delay forwarding downstream of Query through ports that are designated for the other tree vs redundant ports
- Generic Multicast Query
- Group Specific Multicast Queries

All Path double multipath tree with IGMP snooping

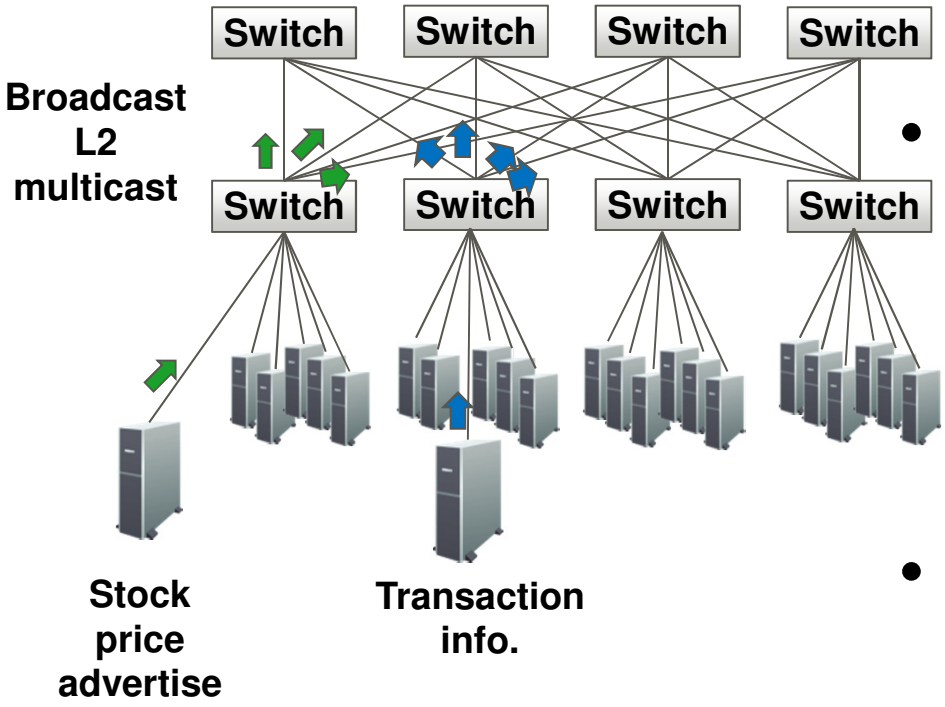
Single multicast tree: All Path vs RSTP with IGMP snooping



- Different pruned trees but similar result:
  - Active branches are selected by **lower latency** in All Path
  - By **nominal costs** in RSTP
- Identical pruning method

Financial cloud: Low Latency

- Multiple sources



- Multiple multicast trees of low latency from multicast source
- Switch generates pseudo IGMP Query packet
- Hosts confirm suscription
- Snooping performs pruning

Conclusions

- All Path belongs to a new line of evolution for bridges (with SRP)
- Multiple chassis implementation seems feasible (ts) plus new mechanisms
- Excellent, native distribution of load when multiple paths with close latencies are available
  - Best suited for campus, enterprise, small data centers than metro.
- Interoperability and scalability
  - **Compatibility of All Path with standard switches at core**
    - No need to develop All Path highest performance switches
  - Makes possible a “sandwidch type” coexistence (RSTP islands-All Path aggregation switches- Standard core switches)
  - Isolated trees of switches at edge of All Path bridges. RSTP at edge islands
  - No shared media: restricted to hosts sharing a link
- Multicast optimization offers interesting performance in financial cloud like (multiple multicast sources) scenarios for low latency multicast distribution
- Protocol diversification continues: several forwarding variants (e.g. per flow path)

FUJITSU

Universidad de Alcalá
Dpto. de Automática
Área de Ingeniería Telemática

La Suma de Todos
Consejería de Educación
Comunidad de Madrid
www.madrid.org

Junta de Comunidades de
Castilla-La Mancha

- Next :
  - Porting NetFPGA implementation to a basic pilot switch for extensive testing, prior to ASIC implementation.
  - A switch platform (preferably FPGA based) is needed
  - Porting to 4*10 Gb NetFPGA board
  - Tests of combined networks (standard and All Path switches)
  - Multicast implementations
  - Contributing to AVB group (all path forwarding, loop prevention)

Thanks

FUJITSU

Universidad de Alcalá
Dpto. de Automática
Área de Ingeniería Telemática

La Suma de Todos
Consejería de Educación
Comunidad de Madrid
www.madrid.org

Junta de Comunidades de
Castilla-La Mancha