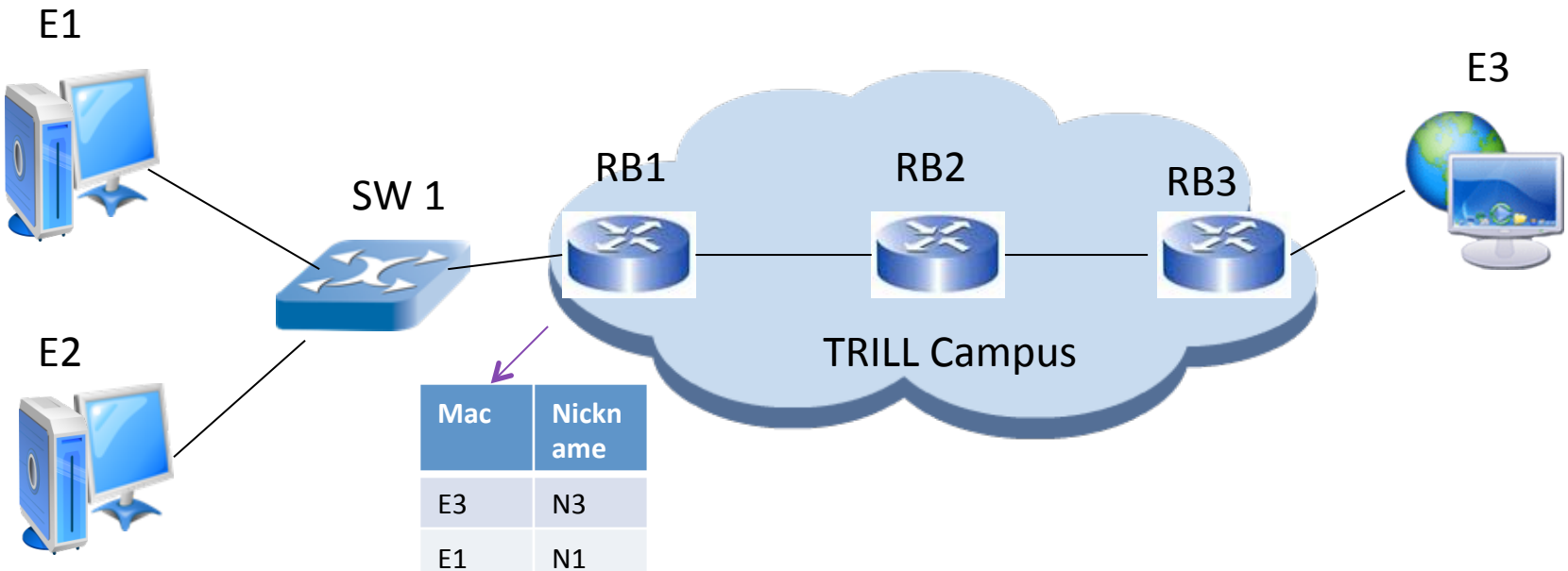
TRILL Smart Endnode

draft-perlman-trill-smart-endnode-02.txt
Radia Perlman
Fangwei Hu
Donald Eastlake 3rd
Kesava Vijaya Krupakaran
Ting Liao

Problem statement

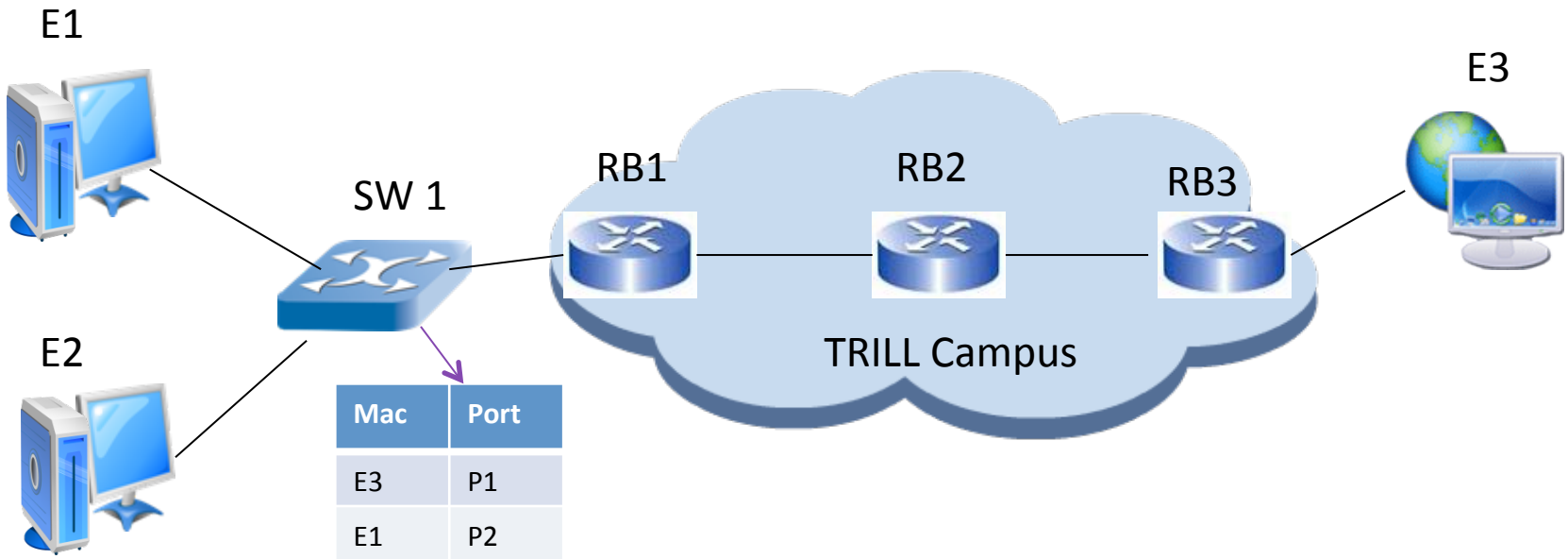
- Edge RBridge RB1
  - Encapsulate and decapsulate TRILL frame
  - Keep endnode learning table (MAC, nickname)
- Table entry
  - Could become very large
  - Be difficult for edge RBridge to notice the changes if endnode move to a different switch



Mac	Nickname
E3	N3
E1	N1
E2	N1

Problem statement

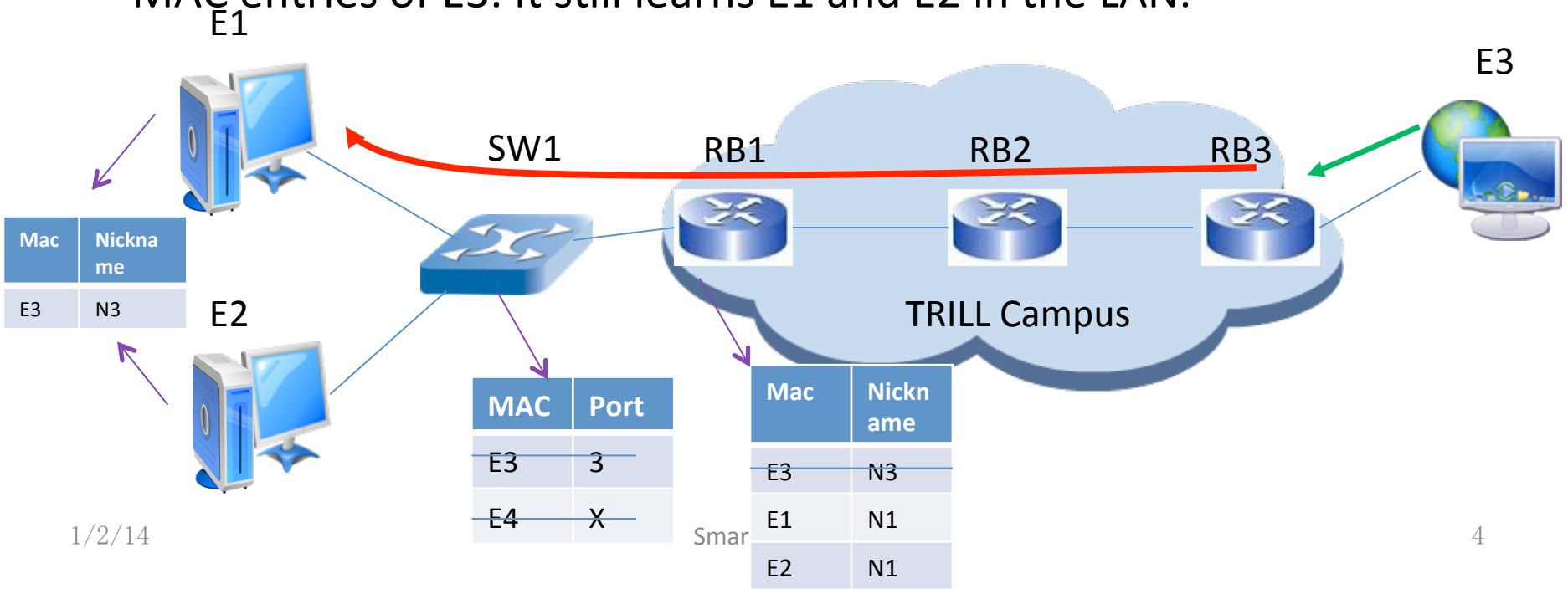
- Switch SW1
  - It learns the MAC address of remote endnode (makes table at SW1 large)
  - If it doesn't know E3, it will flood to all the endnodes



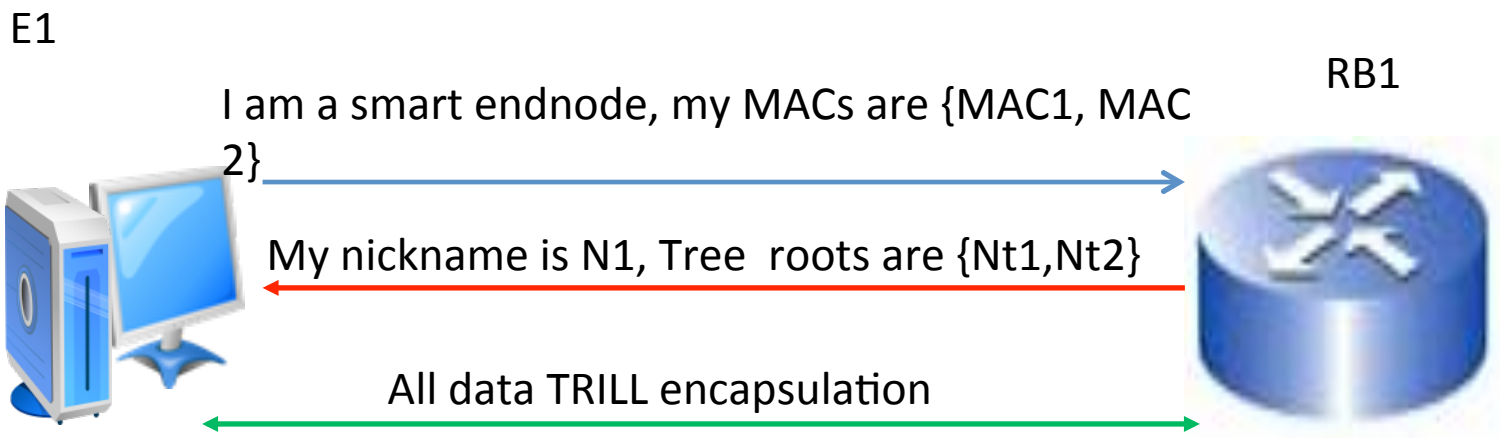
Mac	Port
E3	P1
E1	P2
E2	p3

Proposed solution: “Smart Endnode”

- “Smart endnode(E1)” encapsulates TRILL frame with edge RBridge’s (RB1)nickname;
- RB1 does not keep the (MAC, nickname) of the remote endnode (E3)
- The switch(SW1) in the LAN attached to RB1 does not need to keep MAC entries of E3. It still learns E1 and E2 in the LAN.



TRILL-Hello



1. Keep (MAC, Nickname) pair
2. Encapsulate trill data frame with the source nickname as N1 assigned by RB1

1. Mark E1 as Smart endnode
2. The data to E1 would be kept encapsulated

Smart Endnode

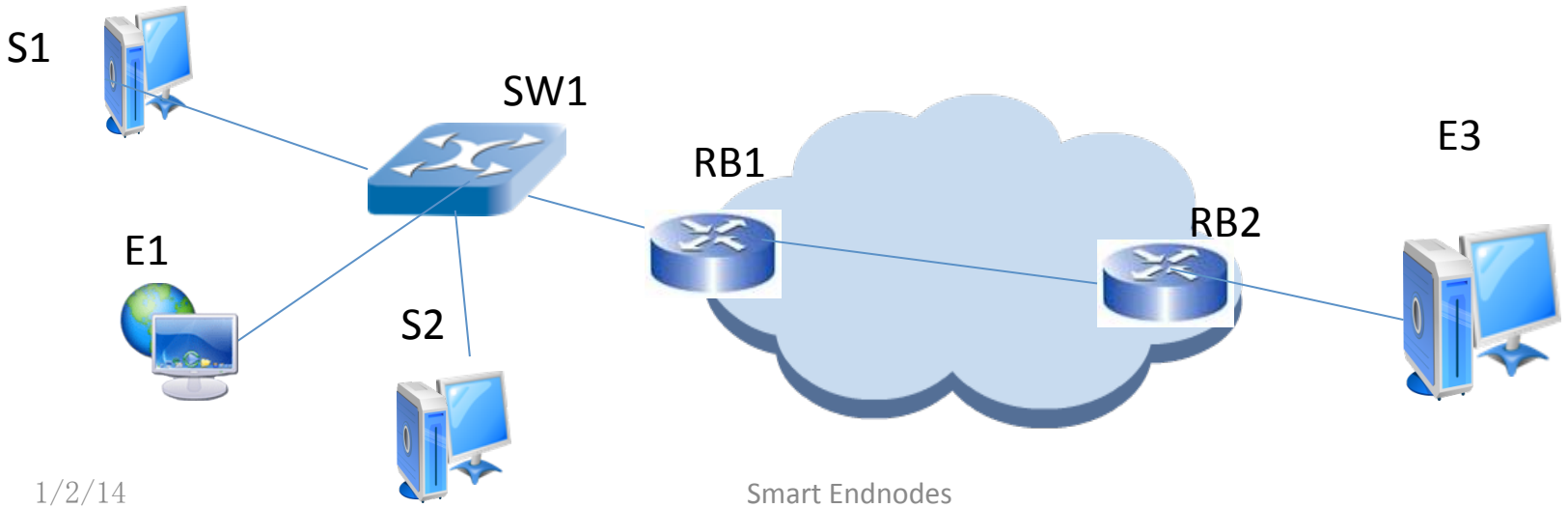
- Does not issue LSPs, nor does receive LSPs and calculate topology
- Sends special smart endnode TRILL-Hello (How often see next slide).
- Maintains (MAC, nickname) table of remote endnodes
- Unicast frame:
  - Destination D is known, ingress nickname is RB1's, egress nickname as indicated in table for D.
  - Destination D is unknown, queries the directory server or use one of the tree nickname assigned by RB1.
- Multi-destination frame: Encapsulates the frame with Nt (one of the root nickname)

When E1 Sends Smart Endnode TRILL-Hello

- When smart endnode(E1) starts up
- Periodically , but not often
- If receives TRILL-Hello from RB1 without mention E1.

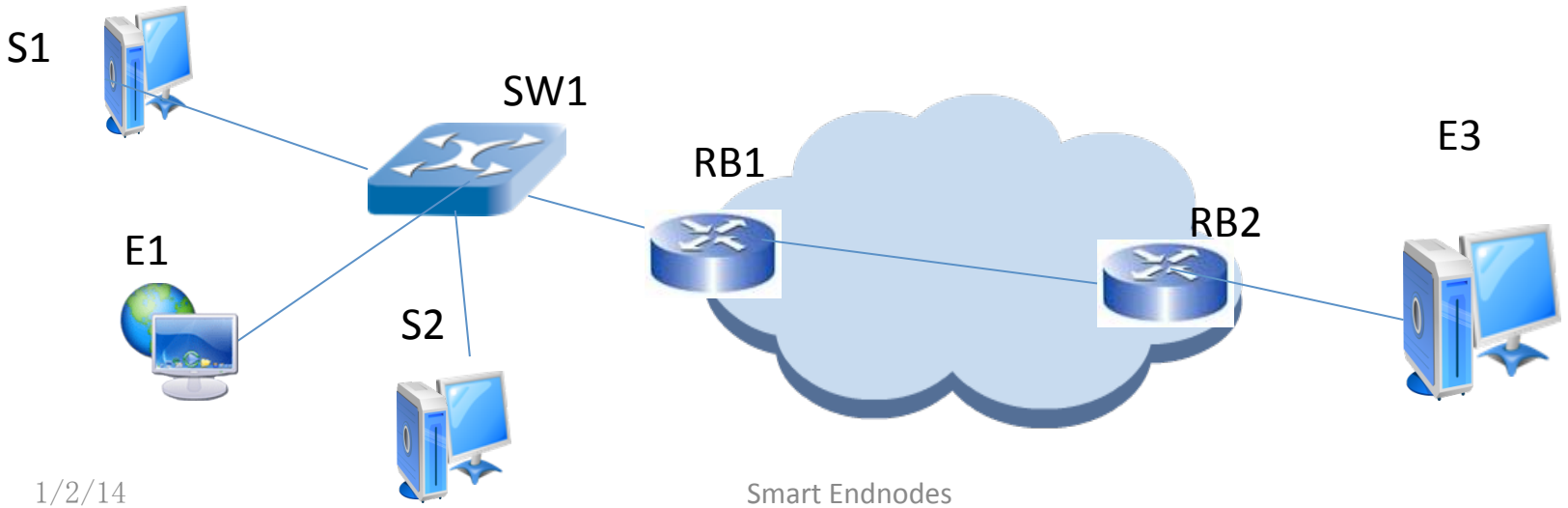
Link to Smart Endnodes

- Unicast from campus
  - If destination MAC is Smart endnode(s1), Keeps encapsulated
- Multicast from campus
  - Forwards to link (encapsulated).
- Multicast to Campus from Smart endnode
  - RB1 Forwards the encapsulated to TRILL campus.
  - Smart Endnode Must listen to “ALL-RBridge” Multicast MAC



Hybrid link:Smart and normal endnodes

- Unicast from campus
  - If destination MAC is Smart endnode(s1), Keeps encapsulated
  - If destination MAC is not known smart endnode (E1), decapsulates
- Multicast from campus
  - sends two copies, encapsulated and native.
- Multicast to Campus from Smart endnode
  - RB1 Forwards the encapsulated to TRILL campus.
  - RB1 decapsulates the frame and back to the hybrid port
  - Smart endnode Must listen to “ALL-RBridge” Multicast MAC



Next Step

- Comments?
- WG adoption?

Thanks!