RPR Topology Discovery Proposal

John Lemon

Italo Busi, Alcatel

Jason Fan, Luminous

Pietro Vittorio Grandi, Alcatel

David James, Cypress

Jim Kao, Cisco

Vittorio Mascolo, Alcatel

Thierry Paiemen, Cisco

Harry Peng, Nortel

Martin Wollensak, Cisco

Necdet Uzun, Cisco

Daniel Zhu, Cisco

Proposed Draft Text

- clause10_topology_jl.pdf

Goals

- Simple
- Tolerant of message loss
- Cause minimal overhead
- Determine/validate connectivity and ordering of stations on the ring
- Ensure all stations on the ring have a uniform and current image of the topology
- Immediate reaction to changes

Goals, continued

- Operate without any master station on the ring
- Operate independently of and in the absence of any management systems
- Support dynamic addition/removal of stations (and ringlets) to/from the ring
- Detect mis-cabling between stations
- Provide means of sharing additional information between stations
- Scalable from 1 to 100’s of stations

Algorithm Basics

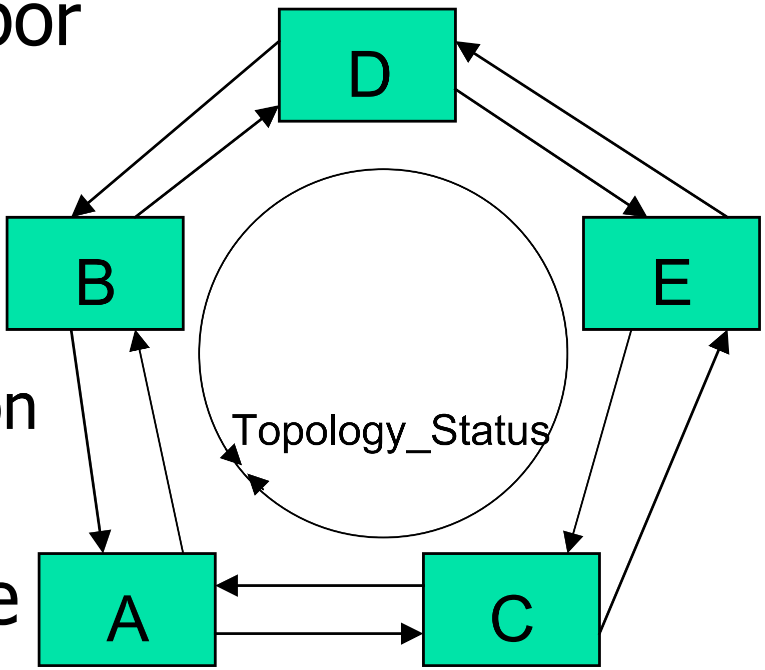
- Each station broadcasts local topology
  - Broadcast periodically, with back off
- Each station announces itself to neighbor
  - One hop “broadcast” periodically, with back off
- Single state state machine, with 5 Events
  - Startup
  - Station status change
  - New neighbor
  - Neighbor and status message timer pops

Information Sharing

- RPR Topology Image used by other algorithms
  - Steering portion of protection algorithm uses Topology Image to know when steering is needed
  - Fairness algorithm uses Topology Image to know where congestion is being experienced (when used with Type 2 messages)
- Several algorithms use station capabilities

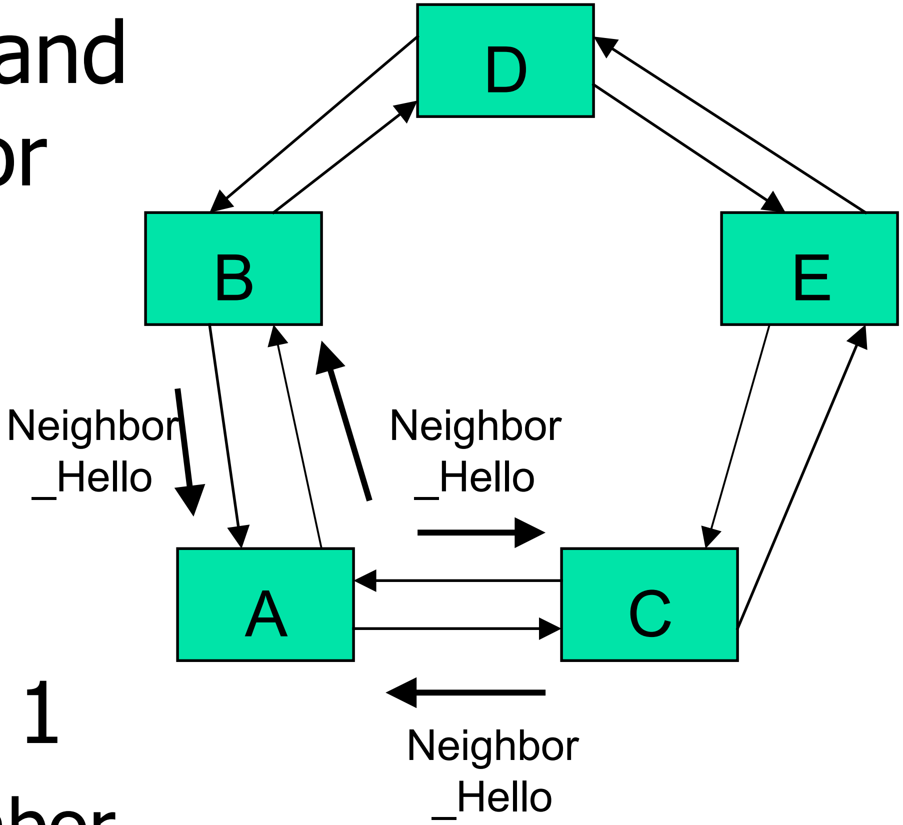
Topology_Status Control Message

- Reports changes in neighbor identity
- Key fields
  - Neighbor MAC addresses
  - Source station image version
- Broadcast on each ringlet with TTL = Max_Ring_Size
  - Removed by source



Neighbor_Hello Control Message

- Reports presence and identity of neighbor station
- Key field
  - Ringlet ID
- Broadcast on each ringlet with TTL = 1
  - Removed by neighbor



State Machine Details, 1

- Startup
  - Trigger
    - Start of state machine
  - Actions
    - Set the local station_image_version to 0
    - Send a Neighbor_Hello on each ringlet
    - Start the Neighbor_Hello_Timer
    - Broadcast a Topology_Status message on each ringlet
    - Start the Topology_Status_Timer

State Machine Details, 2

- Station status received
  - Trigger
    - A Topology_Status message is received
  - Actions
    - If ringlet_id in message matches ID of ringlet on which message was received, then continue.*
    - If station_image_version is higher, then continue.
    - Update local topology image with remote info
    - Broadcast a Topology_Status message on each ringlet (if remote station is new)

State Machine Details, 3

- Neighbor Hello Received
  - Trigger
    - Neighbor_Hello message received
  - Actions
    - If the SA in the Neighbor_Hello is different from the stored neighbor address, then continue
    - Increment local station_image_version
    - Broadcast Topology_Status message on each ringlet

State Machine Details, 4

- Topology_Status_Timer Pop
  - Trigger
    - Topology_Status_Timer Pop
  - Actions
    - Send a Topology_Status on each ringlet
    - Start the Topology_Status_Timer

State Machine Details, 5

- Neighbor_Hello_Timer Pop
  - Trigger
    - Neighbor_Hello_Timer Pop
  - Actions
    - Send a Neighbor_Hello on each ringlet
    - Start the Neighbor_Hello_Timer

Timers

- Topology_Status_Timer
  - Started at initial start of topology discovery algortihm
  - Reset when local topology changes
  - Initial value of 2 ms*
  - Increases by factors of 2 up to 1000 ms
- Neighbor_Hello_Timer
  - Started at initial start of topology discovery algortihm
  - Initial value of 2 ms*
  - Increases by factors of 2 up to 1000 ms

Optimizations

- Station_Image_Version
  - Starts at 0
  - Incremented upon each change in local status
  - Independent value for each station
  - Quick check to see if status has changed from locally stored status
- Reset Topology_Status_Timer only on receipt of Topology_Status from new station

Possible Combinations

- Combination with protection algorithm
  - Topology and protection both giving status of some sort for neighbor links
  - Topology updates needed 3 orders of magnitude less often than protection updates
- Combination with fairness algorithm
  - Topology and fairness both giving status of some sort for neighbor links
  - Topology updates needed 3 orders of magnitude less often than fairness updates