



# Fujitsu 802.17 Requirements

Bob Sultan

[bob.sultan@fnc.fujitsu.com](mailto:bob.sultan@fnc.fujitsu.com)

Ajay Sahai

Sushil Pandhi

Fujitsu Network Communications

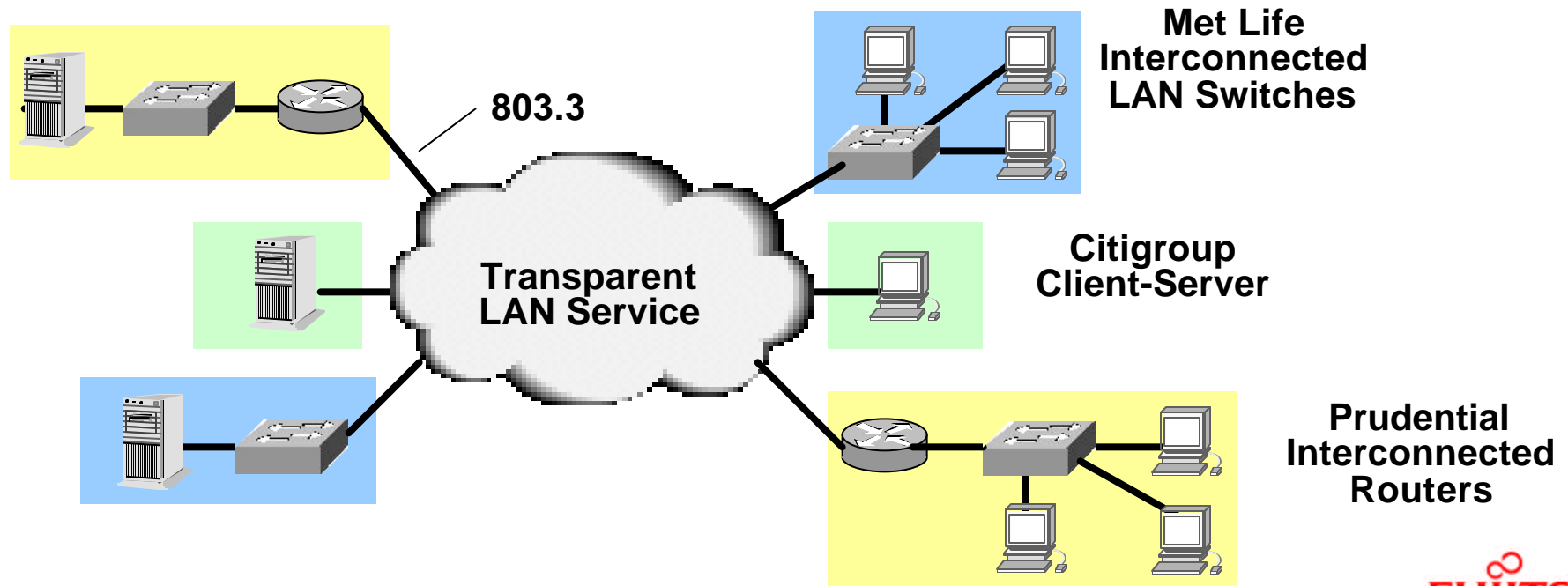


THE POSSIBILITIES ARE INFINITE



# Transparent LAN Service

- users interconnected as if over Ethernet LAN
- LAN 'private' to each customer (customer separation)
- usual Ethernet features (VLAN, priority, flow control)

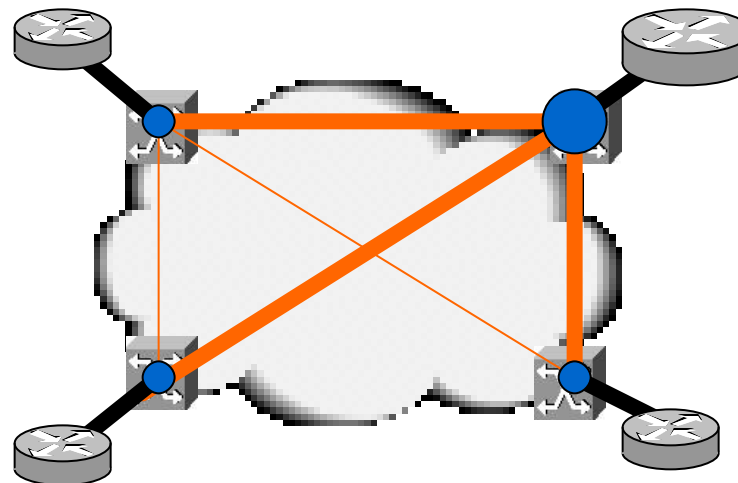
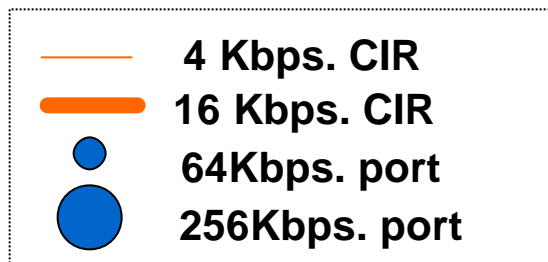


**FUJITSU**



# Frame Relay Service Model

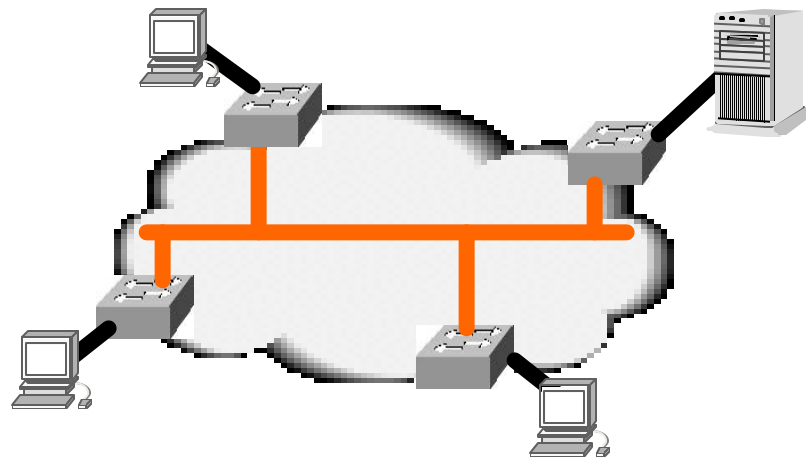
- Port-rate (per port)
- Committed Information Rate (per PVC)
  - Between  $N-1$  and  $N(N-1)/2$  Connections





# Aggregate-Rate Service Model

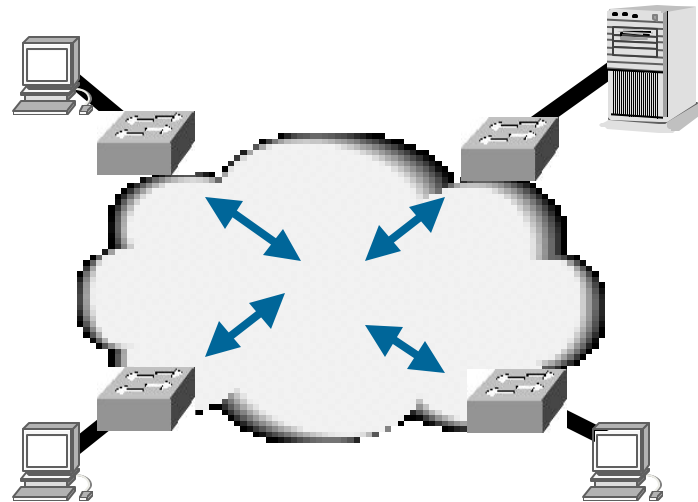
- **Single aggregate rate shared among enterprise users**
  - model as shared medium like collision-based Ethernet
- **Provision one aggregate rate vs. many PVCs**
- **'Collision' occurs when senders' aggregate exceeds provisioned aggregate rate (per customer)**
  - 'collision' does not abort transmission (as in Ethernet)
  - throttle heavy user(s) to reduce future 'collisions'





## Best Effort Service Model

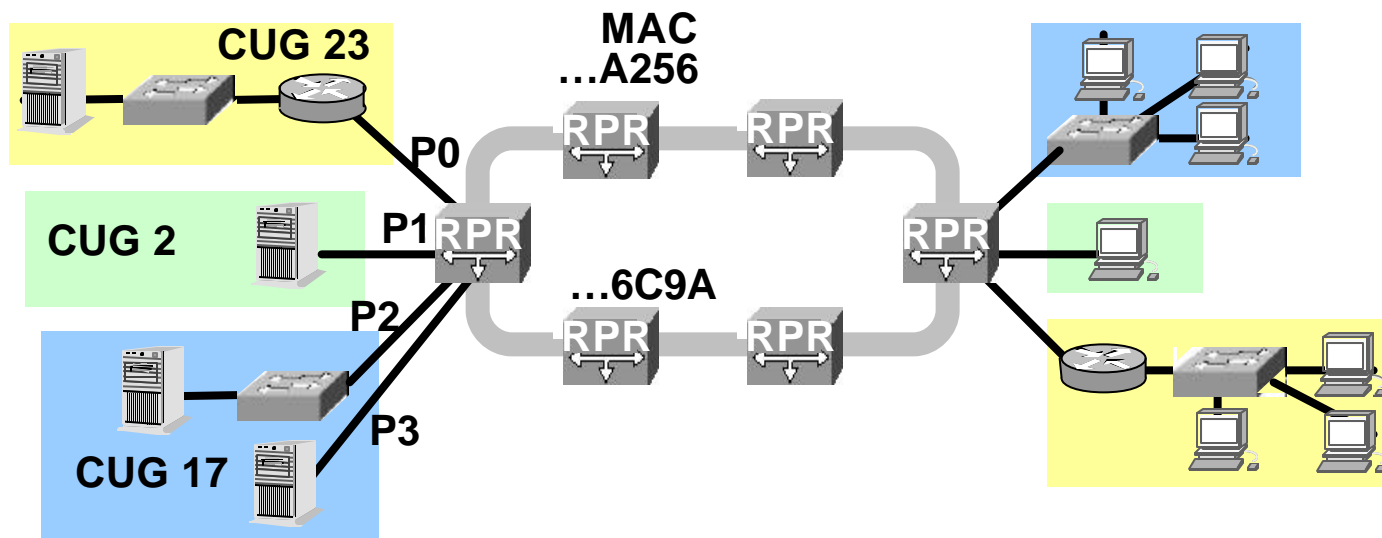
- **No bandwidth guarantees**
  - uses bandwidth available after guaranteed traffic
- **Throttle heavy user(s) when congestion detected**
  - do not penalize well-behaved users





## Transparent LAN Service over RPR

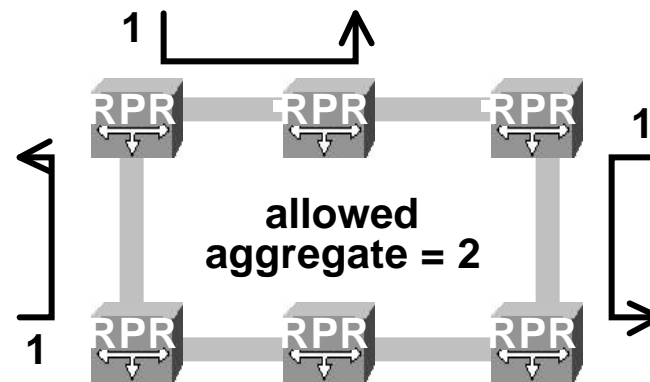
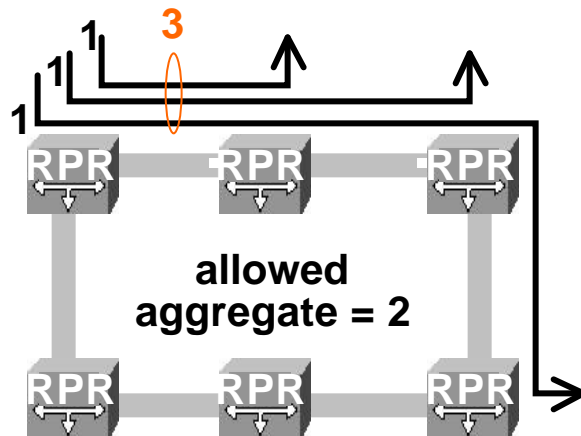
- User 802.3 encapsulated in 802.17
- Customer separation by Closed User Group (CUG)
- Customer port identified by MAC.PORT
  - Allow multiple ports of same CUG per 802.17 MAC@





# Aggregate Rate Guarantee

- Rule: 'provisioned aggregate' exceeded on link implies 'provisioned aggregate' exceeded on ring
  - so monitor for 'provisioned aggregate' exceeded on link
- 'provisioned aggregate' exceeded on ring is not detected if 'provisioned aggregate' not exceeded on link
  - but allowed as the resource commitment is unaffected

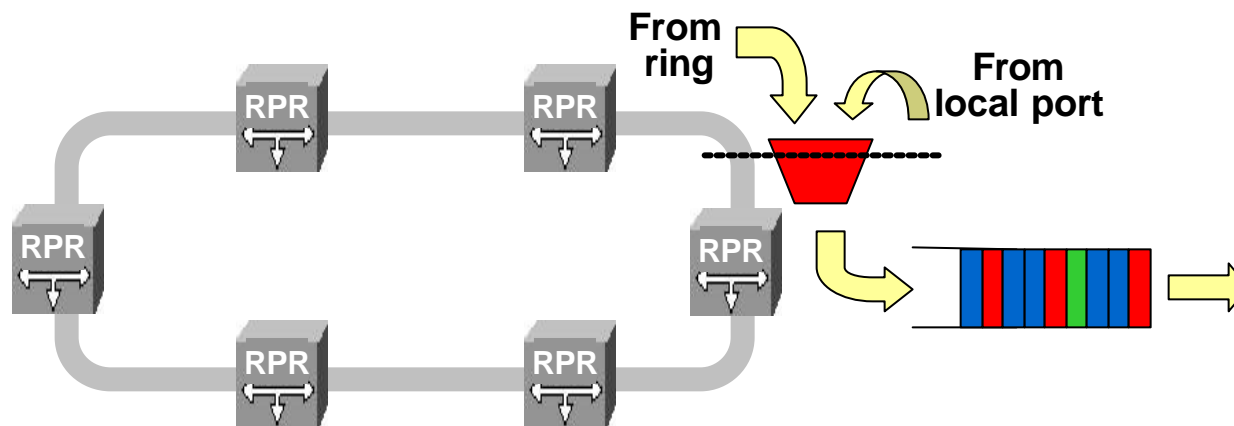


FUJITSU



# Aggregate Rate Guarantee Model

- Leaky bucket *per CUG* on transit and local traffic
  - enforce aggregate rate
- Detect threshold crossing on leaky bucket
  - throttle individual sender within CUG
  - throttle alternatives
    - similar to 802.3 PAUSE (but unicast)
    - request to reduce ingress rate (followed by growth)
- Place frame on guaranteed-service queue

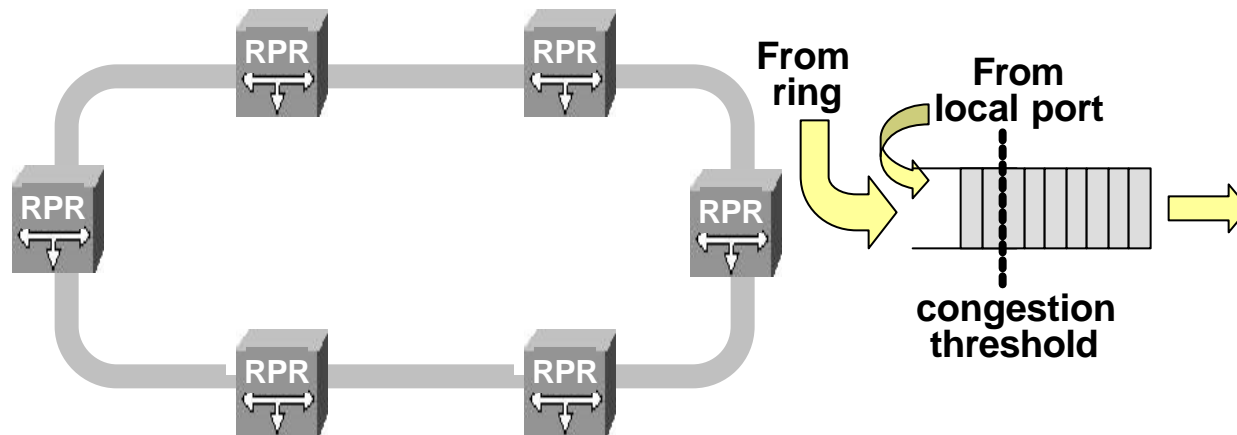






# Best Effort Model

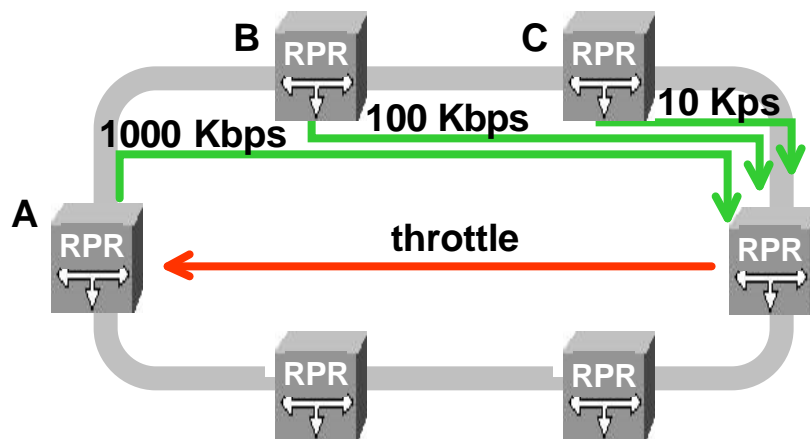
- Place frame on best-effort queue
  - Drop frame when buffer unavailable
- Detect threshold crossing
  - throttle individual sender to avoid buffer depletion





# Choosing Source to Throttle

- **Cache most active senders**
  - including local ports
  - per CUG for guaranteed service
- **Choose (e.g. rotation) among most active sources**
  - Send THROTTLE control message to one source
  - THROTTLE other sources after time interval as needed



rate cache (per CUG)

sender	rate
A	1000
...	...
...	...
B	100
...	...
...	...



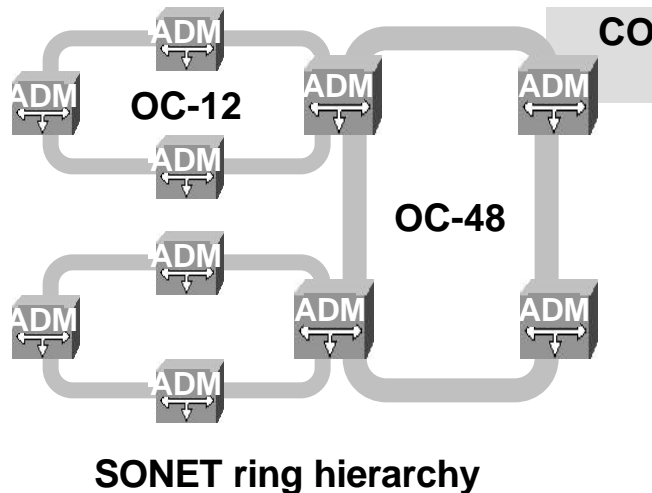
C

FUJITSU

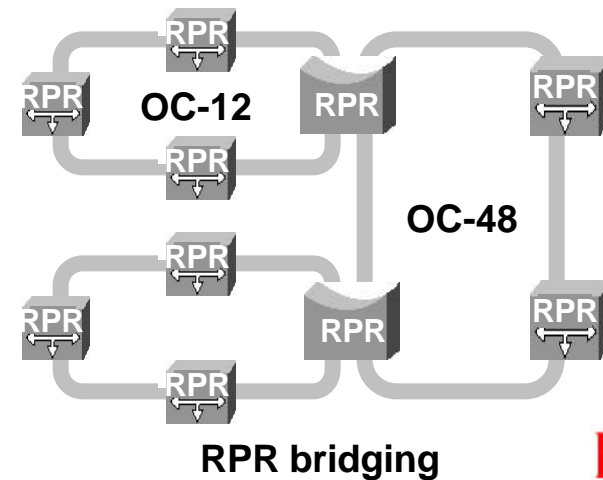


## RPR Ring Interconnect

- **Hierarchical SONET ring topologies deployed today**
  - Strong requirement to extend RPR across rings
- **Tree topology sufficient since rings are protected**
  - STP not required
- **Assume RPR bridge inherits properties of 802.1D**
  - make exceptions/additions as needed



SONET ring hierarchy



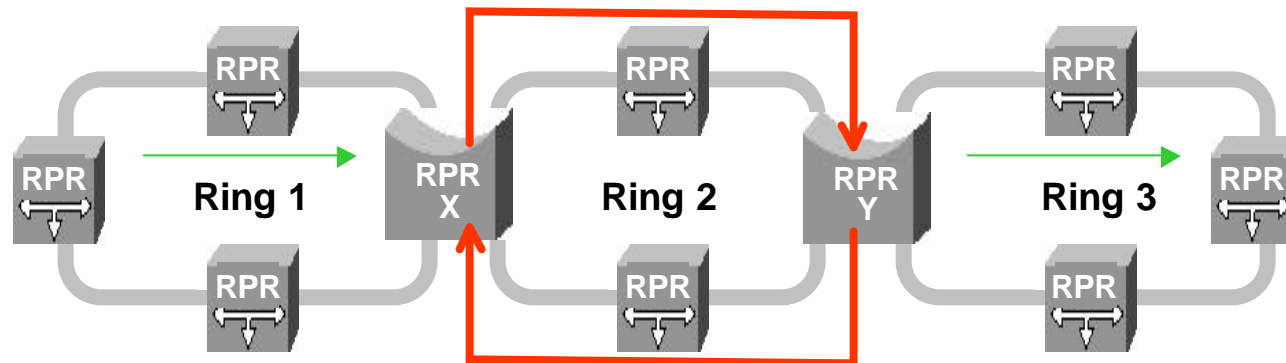
RPR bridging

**FUJITSU**



## Stripping Broadcast Frames at Ring Ingress Bridge

- Broadcast frames stripped at source RPR node
- Bridge X forwards broadcast from ring 1 to ring 2
- Bridge X must strip frame after circulation
- Frame carries ingress Bridge ID for recognition





# Caching Egress Bridge Identity

## ■ Bridge X can learn that

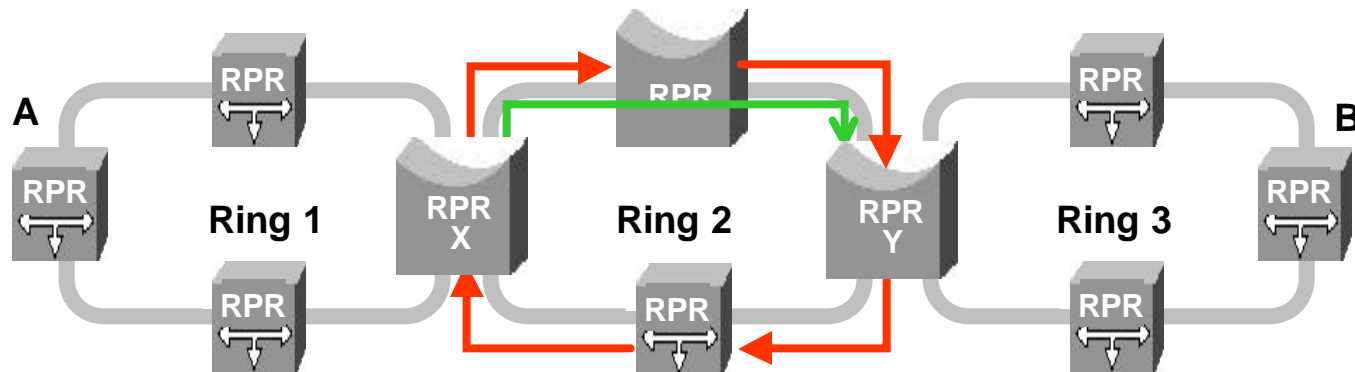
- frames for destination B reached via ring 2
  - bridge X sends frame 'circulate to all bridges on ring'

*OR*

- frames for destination B reached via bridge Y
  - bridge X sends frame on ring 2 with Bridge Y ID

## ■ Directed bridging allows spatial reuse

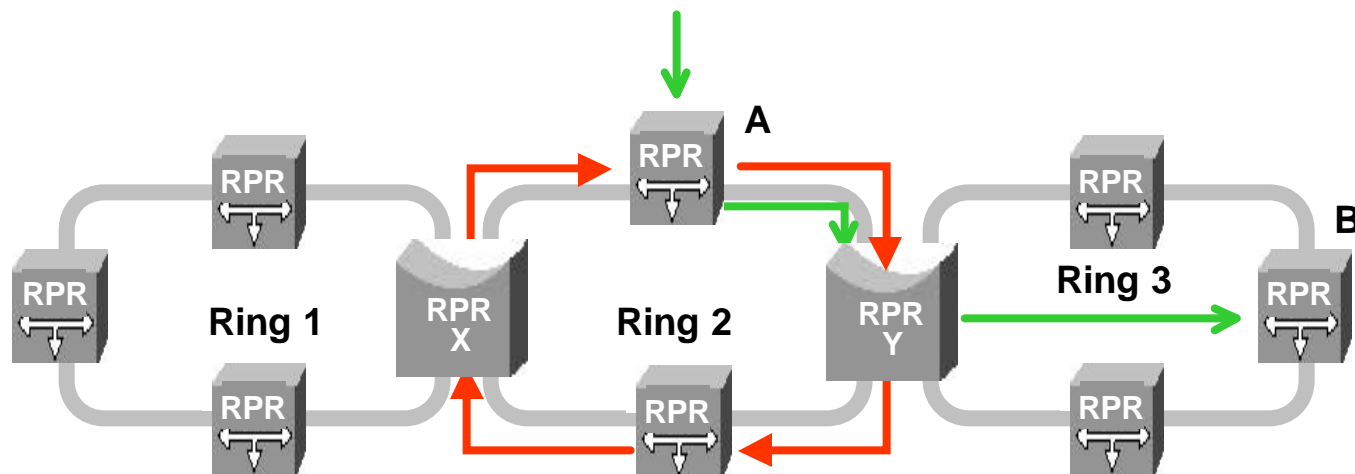
- implies that MAC frame carries identity of egress bridge





## Caching Egress Bridge Identity at Source

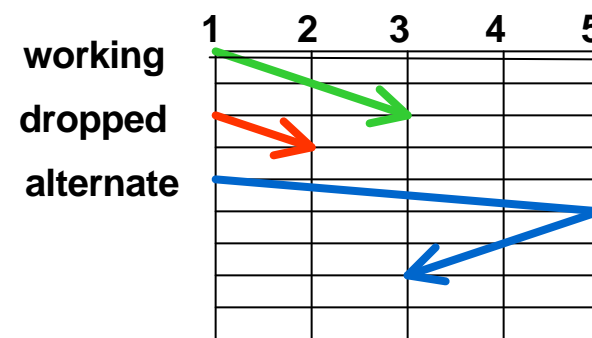
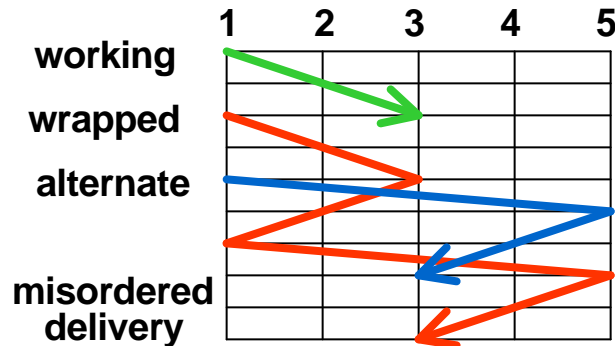
- RPR host node A can
  - circulate frames for destination B (not on local ring)
  - OR
  - direct frames for destination B to bridge Y
- Directed bridging allows spatial reuse
  - implies MAC frame carries identity of egress bridge
  - bridging is *not* transparent to host node





# Protection

- Steering -> dropped frames during recomputation
- Wrapping -> period of disorder on switchover
- Wrapping benefit when recomputation time is long
  - Our assumption that recomputation time is short
  - Failure notification broadcast speeds recomputation
- Steering preferred
  - appears generally simpler





# Frame Format

- **Generally consistent with GFP**
  - Source / Destination MAC (802 MAC)
  - TTL (8 bits)
  - priority (3 bits)
- **Additional or modified**
  - Source / Destination Port Identifiers (8-bit)
    - 4-bit GFP port insufficient
  - CUGID (16-bit)
  - Ingress / Egress bridge identifier (MAC@)
    - reserved value for 'all bridges on ring'





## No Requirement for the Following

---

- Jumbo Frames (encapsulate 802.3)
- TDM / circuit emulation (use SONET)
- Different link speeds between nodes
- Explicit support for 802.1D/Q
  - subscriber 802.3 traffic is encapsulated
  - RPR supports functions *analogous* to 802.1D/Q/p
- Spanning Tree Protocol (limit to tree topology)



# RPR Requirements

- **Physical Layer Mapping**
  - OC-3 to OC-192 link rates
  - NxSTS-1 and NxSTS-3c Virtual Concatenation
- **Interoperability within ring**
  - standard not meaningful without node interoperability
  - essential for low-cost RPR hardware
- **Interoperability between rings (bridging)**
- **Protection Protocol (steering if < 50ms.)**
- **Topology Protocol**
- **Guaranteed and Best-Effort Service Classes**



# RPR Requirements

## ■ Customer separation

- no data leakage among customers
- dynamic registration of CUGs
  - analagous to VLAN registration (reuse GVRP)

## ■ Bandwidth isolation among customers

- absolute for guaranteed-service
- approximate for best-effort

## ■ Bandwidth management for Guaranteed Service

- ingress policing of 'provisioned aggregate' per CUG
- link policing per CUG (support aggregate-rate SLA)
- rate threshold per CUG (congestion control)

## ■ Congestion control

- source rate cache (per CUG for guaranteed service)
- PAUSE or source-rate reduction request



## Summary

---

- **RPR supports aggregate-rate service model**
  - significantly reduced provisioning
- **Enforce aggregate-rate on ring by enforcing on link**
- **Source throttle for congestion control**
  - sender rate caching
- **Ring interconnection is key requirement**
  - base is 802.1D but significant changes required
  - spanning tree not needed
- **Reuse VLAN concepts for CUG**