

# **RPR Traffic Management**

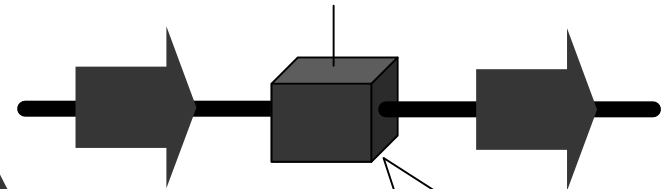
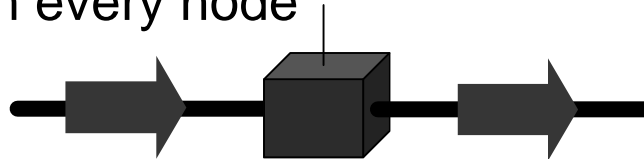
***Distributed Cut-through Switching***

**Frederic Thepot [fthepot@dynarc.com](mailto:fthepot@dynarc.com)**

**Lars Ramfelt [larsh@dynarc.com](mailto:larsh@dynarc.com)**

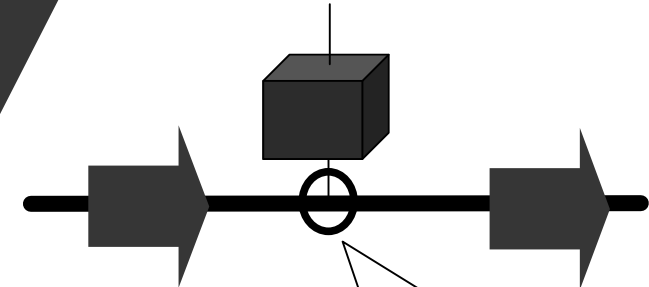
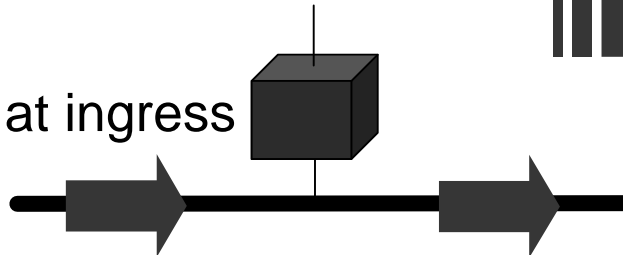
## Scalability with Cut-through Switching in the Ring

Scheduling in every node



Switch must be upgraded

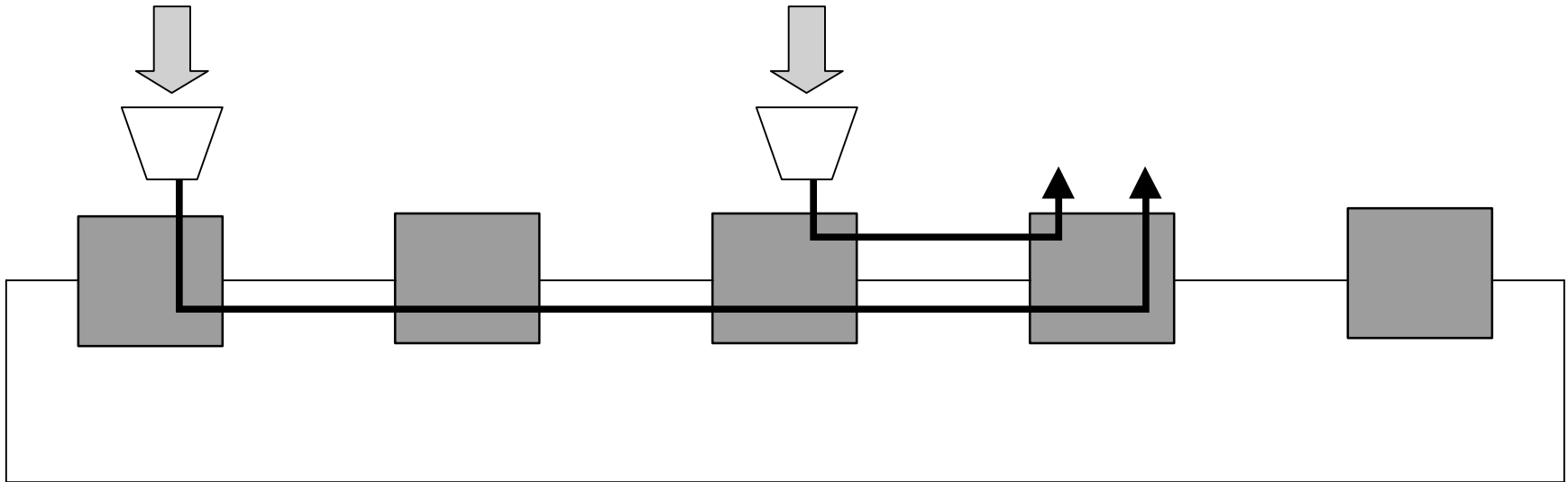
Scheduling at ingress



Interface cards must be upgraded or changed

***A packet add-drop switch with ingress scheduling is the simplest RPR device able to scale in speed and size.***

## RPR: DISTRIBUTED traffic management

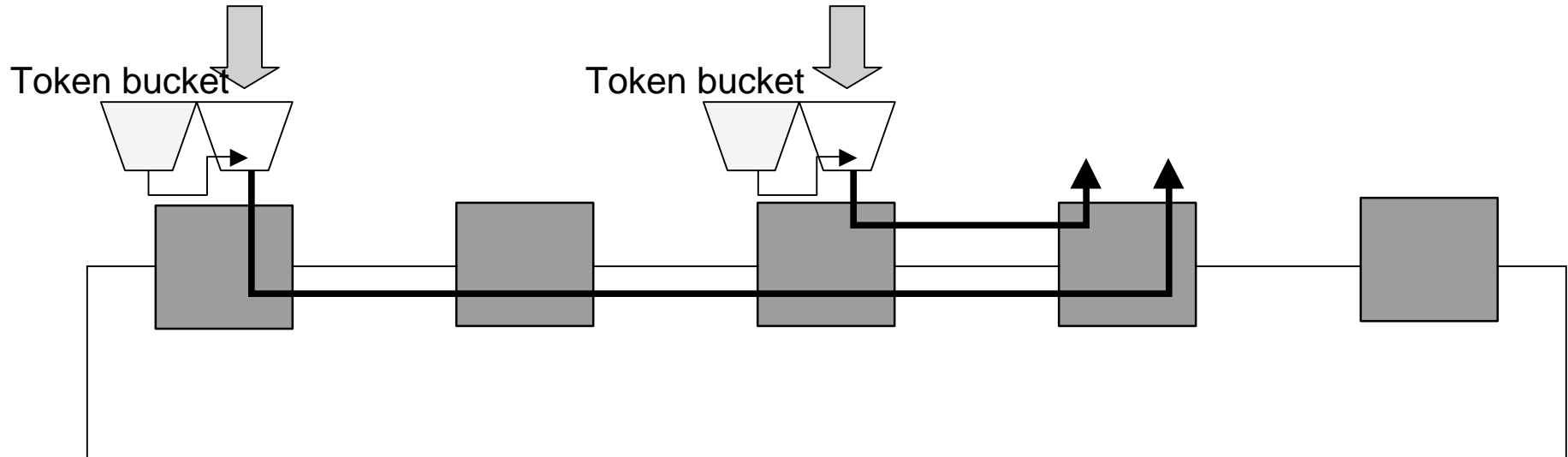


- Scheduling happens only at ingress
- No rescheduling at intermediate nodes!
- Asymmetric model: switched ports, ring ports
- Simple and inexpensive buffer management

## Traffic Management Requirements

- Traffic Schemes are generally asymmetrical
- Sum of node traffic flows gives forecasted aggregated needs
- Possible bandwidth allocation between nodes done dynamically on the fly
- Support for IP DiffServe
- Simple provisioning
- Distributed Scheme that can scale

## RPR: DISTRIBUTED token bucket



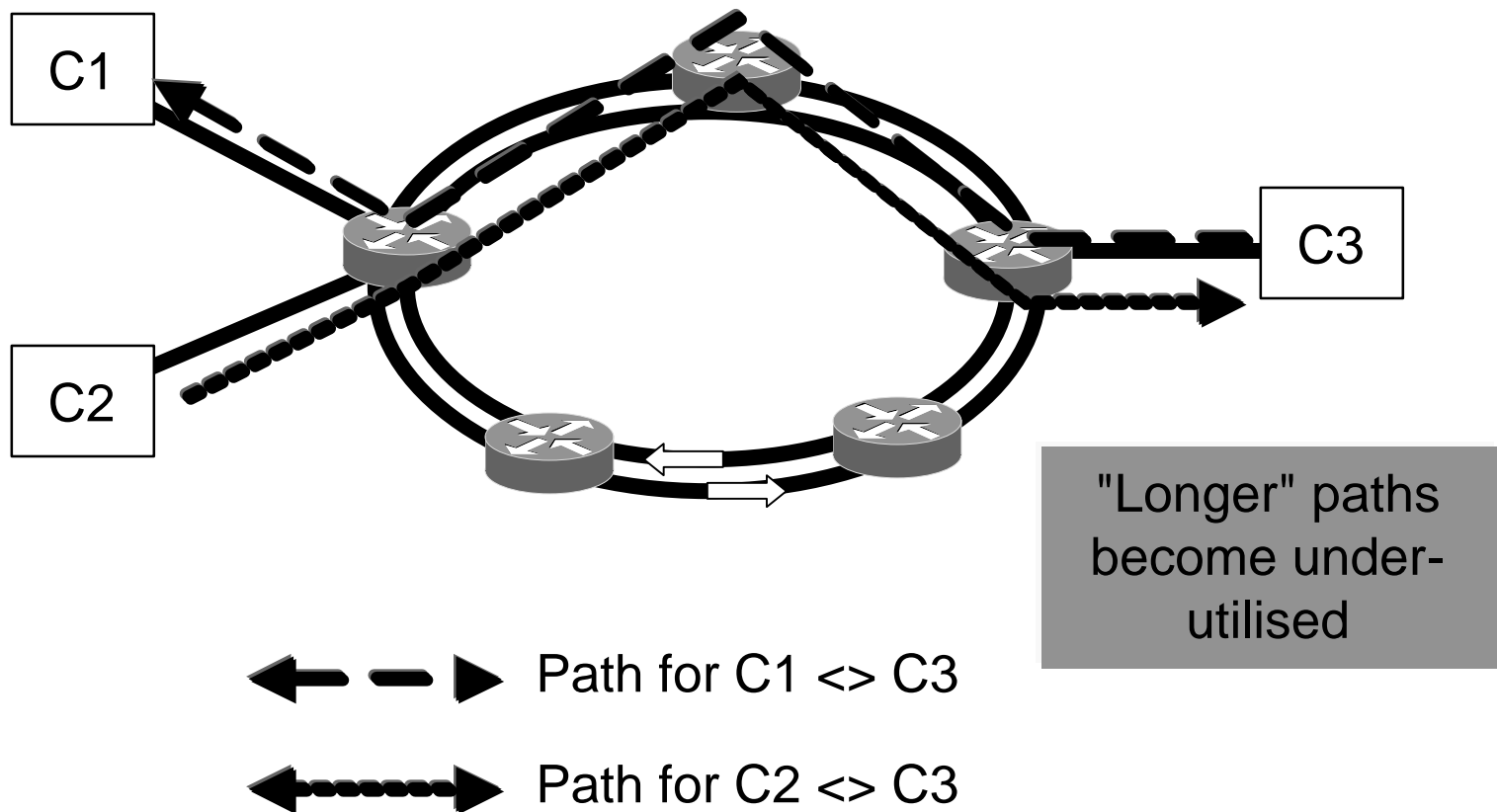
- Scheduling is achieved by a token bucket
- You can send when there is no traffic on the ring port and you are given a token
- A token correspond to a given bandwidth: 1Mbps
- Token Distribution is part of the control plane

## RPR response to congestion

- Congestion in a node on the ring can be remedied immediately by reallocating spare bandwidth with the control plane through token distribution
- If a node is underdimensioned regarding its throughput
  - Another node can be introduced next to it on the ring
  - The backplane can be upgraded to double its processing capacity
- If the concerned ring is close to its maximum throughput
  - The ring can be segmented into two rings with maximum throughput in each
  - The link speed can be upgraded on-line by replacing the interface cards, thus increasing the ring throughput with a proportional factor

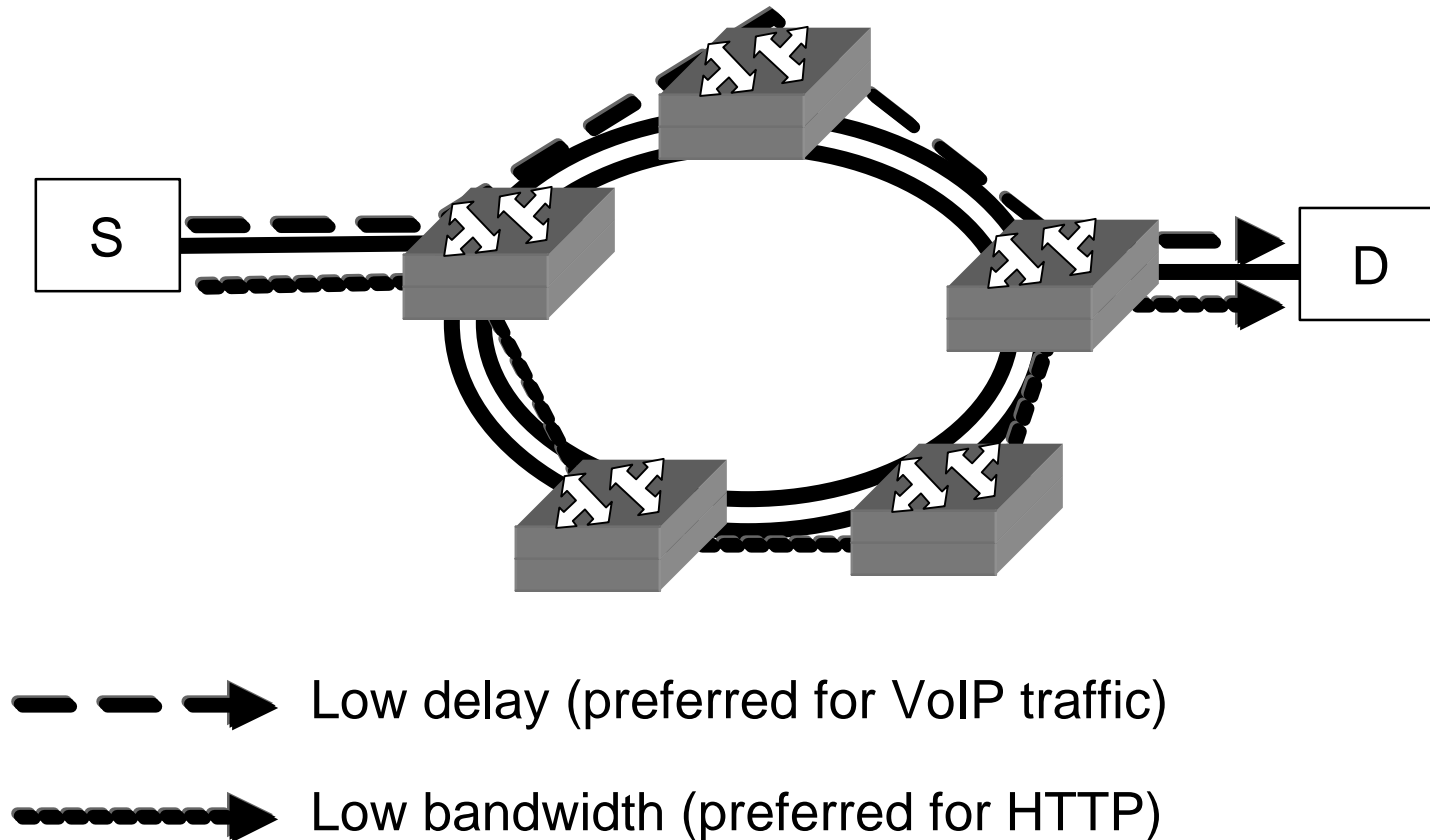
## Today Bandwidth Bottlenecks

Today routing protocols create a single "shortest path"



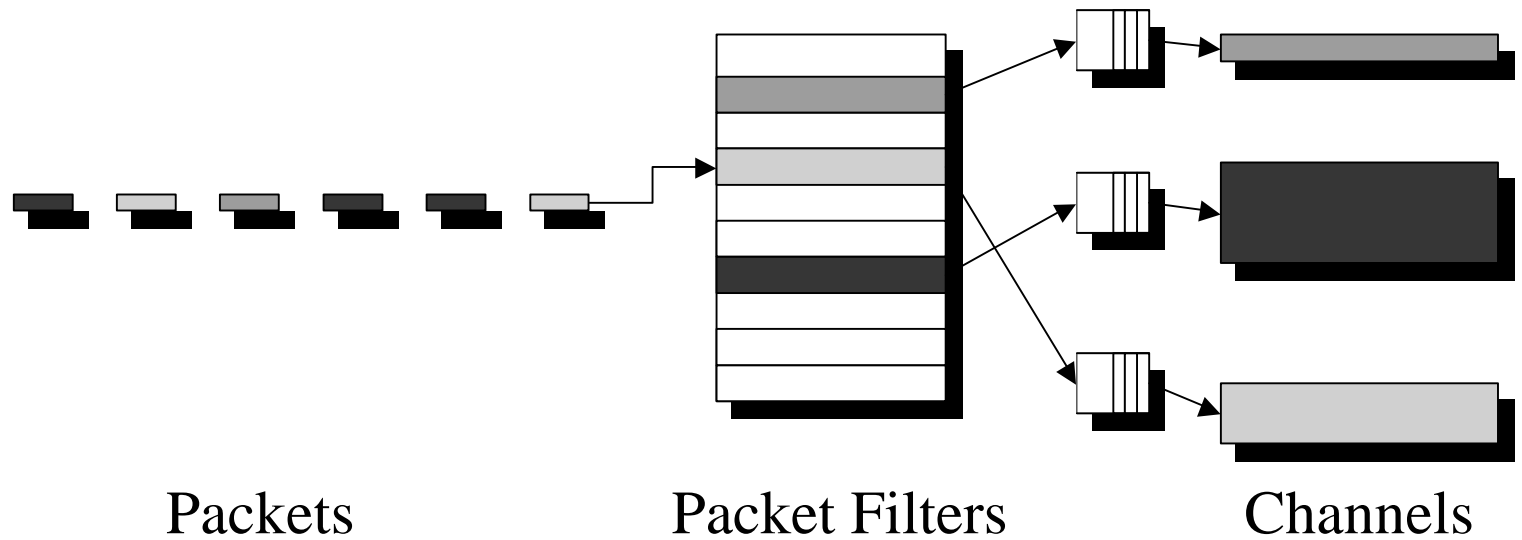
## RPR Ring Awareness

Source device determines the type of path on the basis of the service





## Packets, Filters and Channels



- Packet filters are mapped to resource reservations (channel specifications)
- Channels created on demand (first packet arrives)

## RPR Channels

- Three variations: Unicast, Multicast, Broadcast
- Created by a control message
- Control channel is bi-directional
- Switch nodes create channels on behalf of the sender
- A multi-hop channel must get OK from every switch along its path
- Portion of the bandwidth
- Synchronous between the sender and the receiver
- Virtually no loss of data

## Token Distribution

- Initial distribution at boot process
- Fairness distribution between nodes for token pool
- Initial distribution is SRP ignorant:  $\text{bandwidth} / \text{nodes}$
- Each node use token pool to allocate token to channels
- Channels are SRP aware: allocated token is local and not global
- Local token is point to point
- When token pool is used then token requests can start from other token pool