

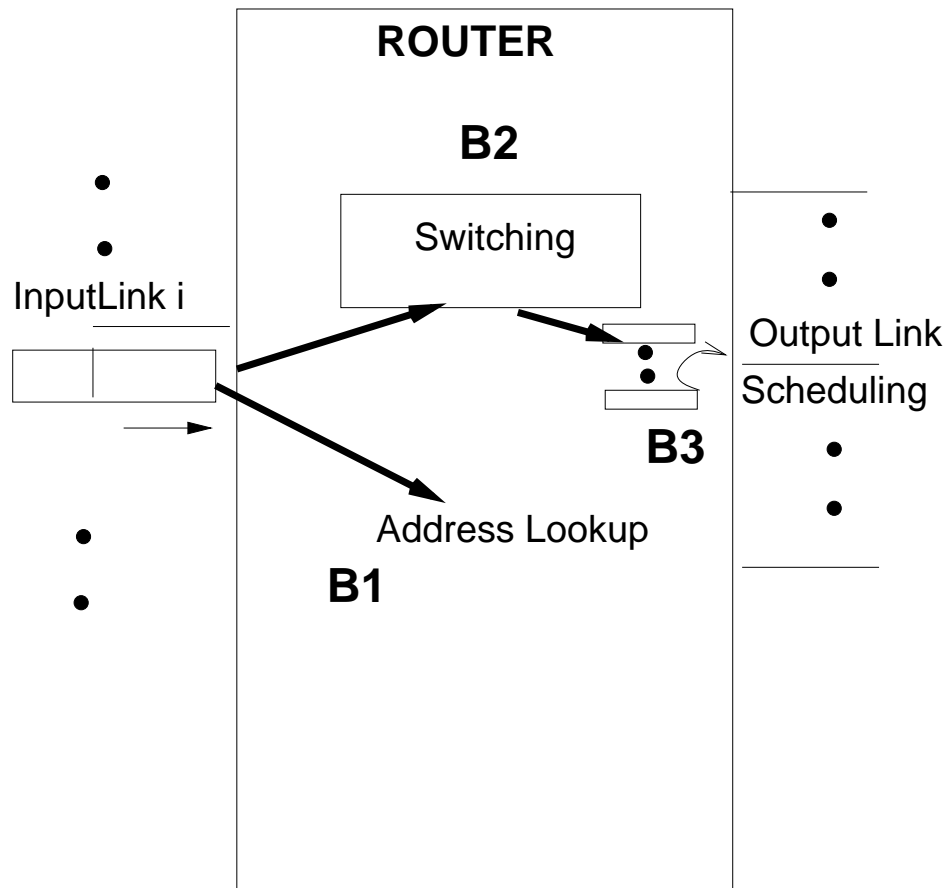
Basic QoS Mechanisms used in Routers

George Varghese

March 5, 2002

Where is Router QoS Implemented

in



- Beyond FIFO. Priority and Round Robin. QoS, delay bounds.
- Output scheduling in real-time path.

QoS is Router Scheduling plus Admission Control

Two parts to this presentation:

- **A.** Basic Router Scheduling Mechanisms (DRR, RED, Priority, Token Bucket, WFQ etc.)
- **B.** Basic Admission Control Mechanisms: RSVP and the Diff Serv Framework.

Why QoS

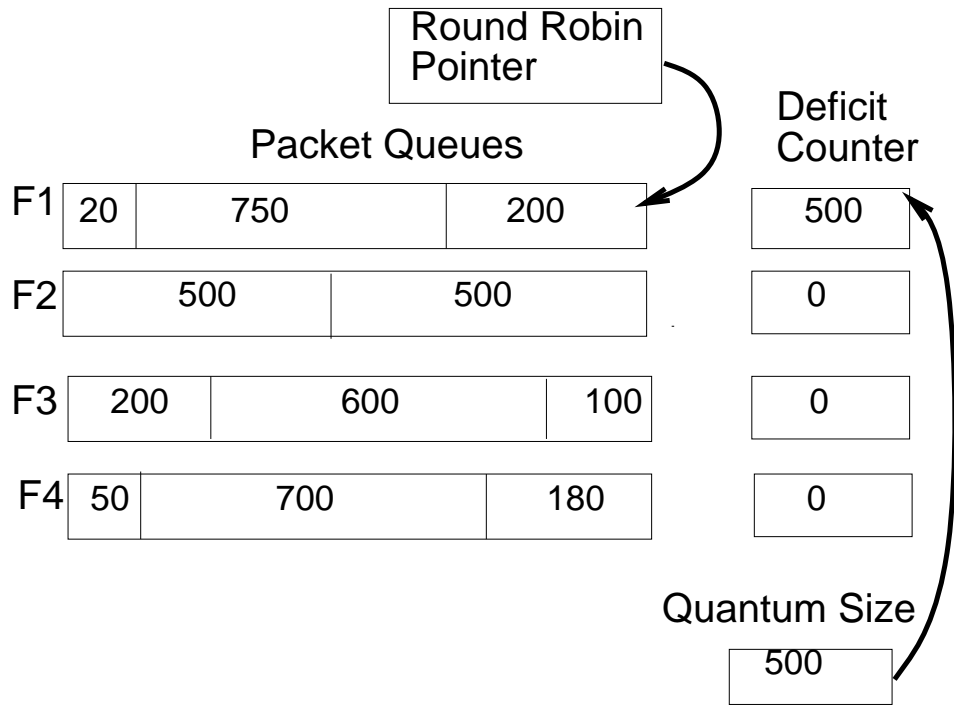
- Improve reaction of TCP sources to congestion.
- Fair sharing among competing flows.
- Provide QoS guarantees to flows (e.g., throughput in VPNs, delay for video) to flows.

Not surprising when one looks at an OS: which load to shed when congested, time sharing, and delay bounds for movie players.

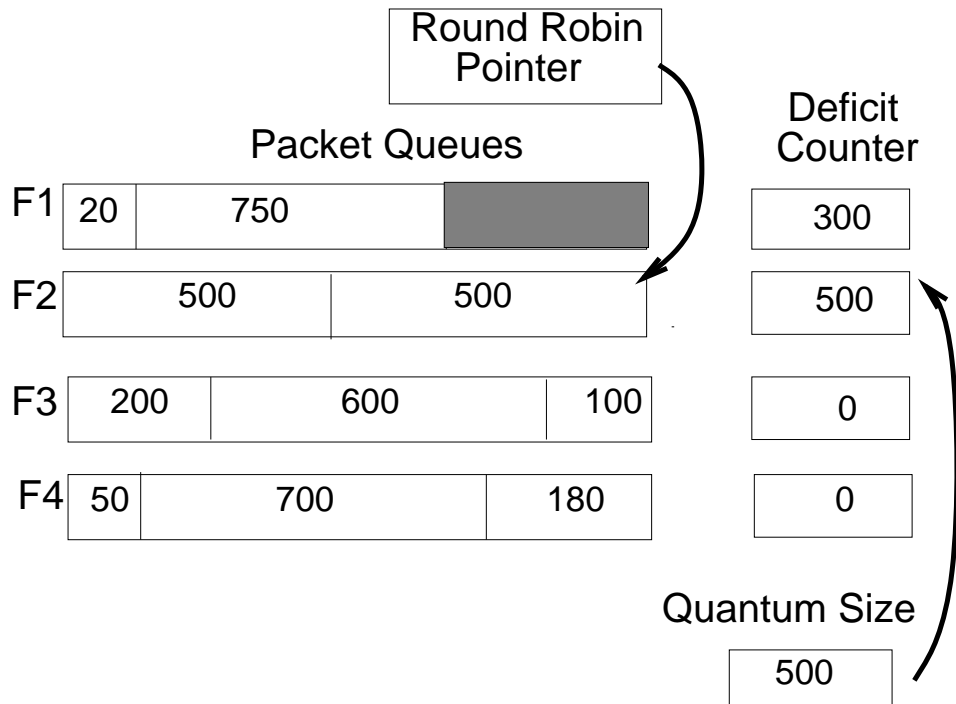
Basic Scheduling Mechanisms used by Routers

- *1.0 Fair Bandwidth Sharing at Output Links using Multiple Queues:* Weighted Round Robin, Deficit Round Robin, and Modified DRR
- *2.0 Fair Queuing at Output Links with Delay Bounds and Bandwidth Sharing:* Weighted Fair Queuing, W2FQ, Virtual Clock, Leap Forward Virtual Clock.
- *3.0 Congestion Avoidance before an output queue:* Random Early Detection and Cisco's WRED for early detection of congestion
- *4.0 Traffic Shaping using Leaky Bucket Controls:* For shaping traffic to meet profiles.
- *5.0 Strict Priority Scheduling:*
- *6.0 Other Topics:* Delay Bounds and Core Stateless.

1.0 DRR and Fair Sharing



1.0 DRR and Fair Sharing

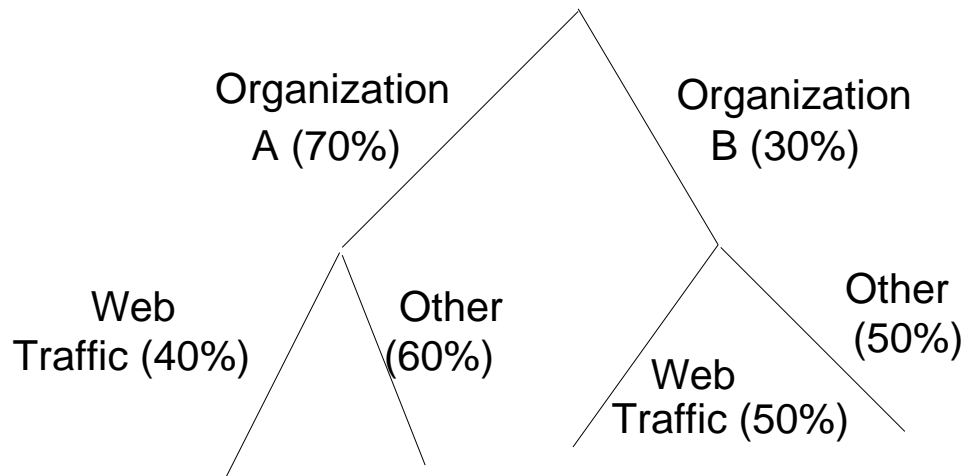


DRR Tricks

- Needs a quantum of at least one maximum packet size to ensure that it does not keep polling queues. Other priorities are formed by larger multiples of packet size.
- To avoid skipping empty queues, we keep an *Active queue* of queue positions that have at least one packet. On servicing, if queue is empty remove it from Active queue, else add it to end of Active queue. Pick next element in queue to service next.
- Storage for active queue and deficit counter can be reduced by more tricks: bit map of active queues with a summary bit, and using randomization for deficit counter.

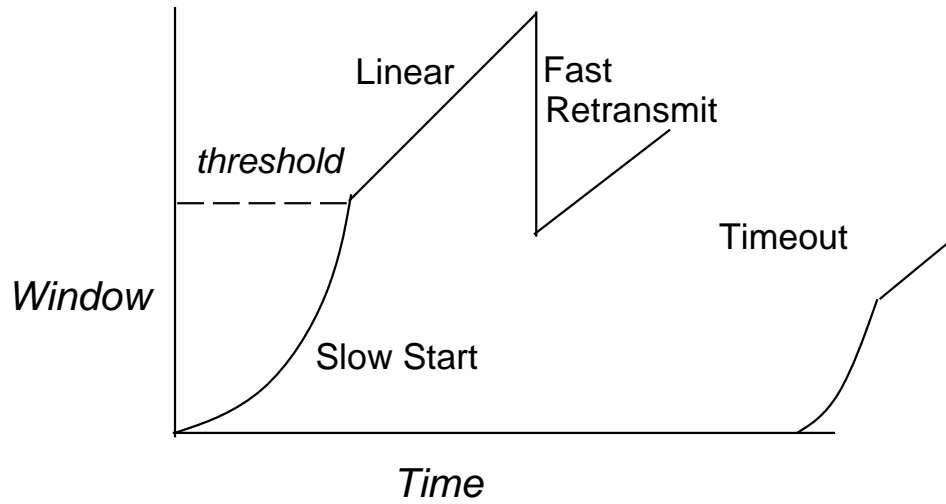
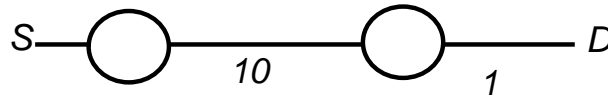
Class Based Queueing

in

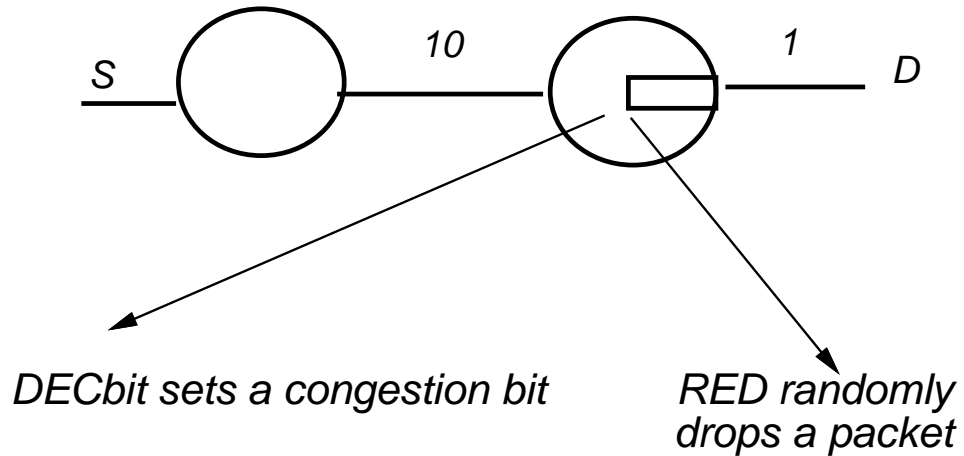


- Can implement using a hierarchical DRR scheduler: one scheduler per node.

3.0 RED and TCP Congestion Control

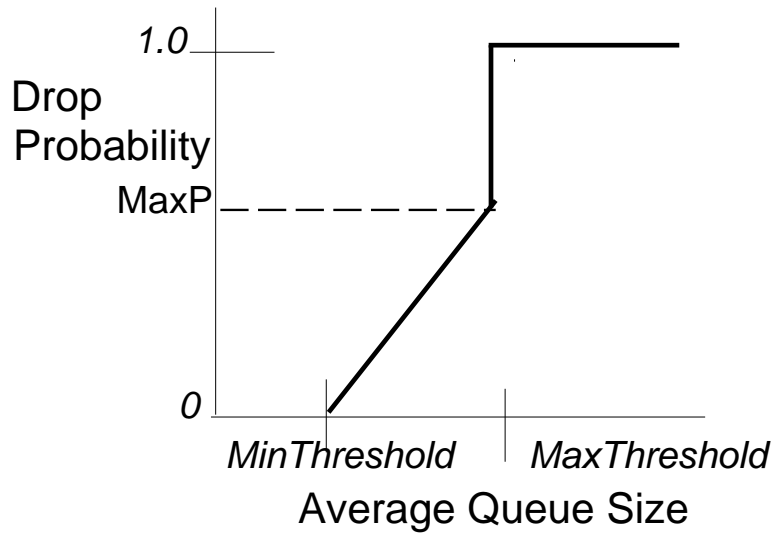


Warning Sources of Impending Congestion



- Today's IP has no room for a congestion bit, so they randomly drop packets with small probability when queue size passes a threshold (RED).
- Proposal on table for a ECN bit for IPv6.

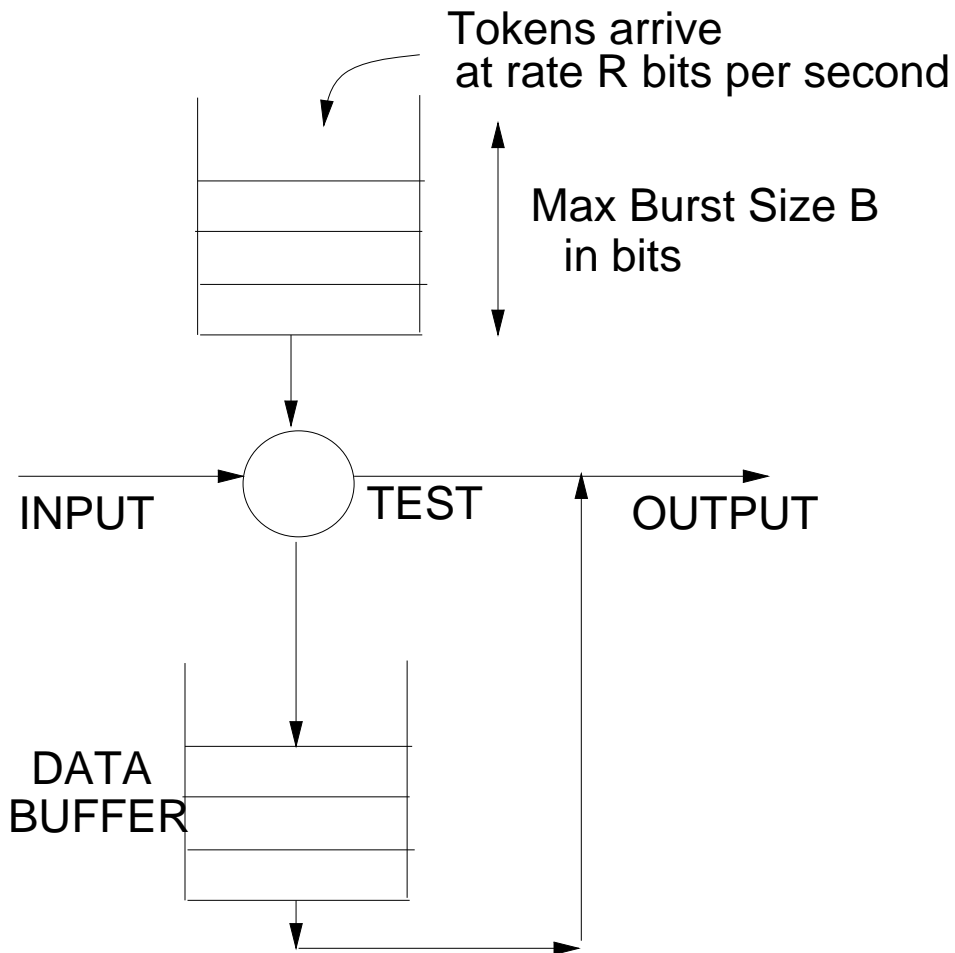
RED in more detail



$$\text{AverageQ} = (1 - w) * \text{AverageQ} + (w * \text{SampleQsize})$$

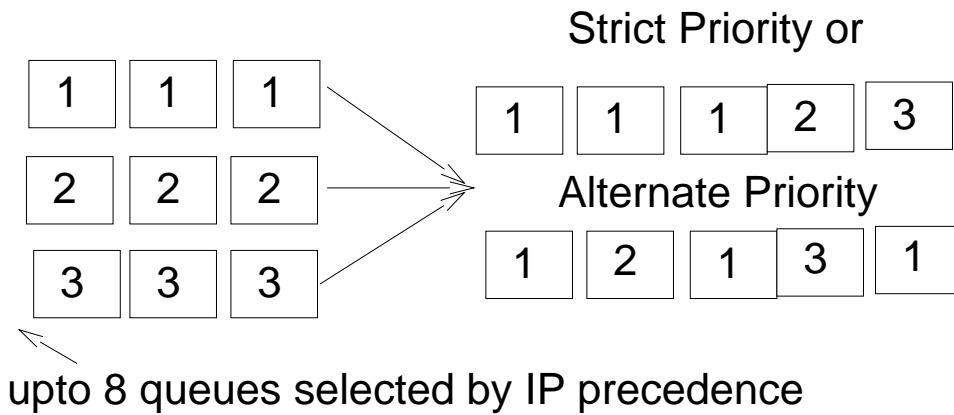
- More complicated by need to keep track of time for which queue is idle. Weights can be powers of 2.
- Cisco introduced Weighted RED where the thresholds can vary depending on type of traffic decided by IP TOS bits.

4.0 Token Bucket Shaping



- Uses: limit traffic type injected to peer (e.g., NEWS), avoid UDP congestion from central server to slow remote line, subrate service.
- Can become a policer (remove buffer) or can simply mark out-of-profile packets.

5.0 Priority

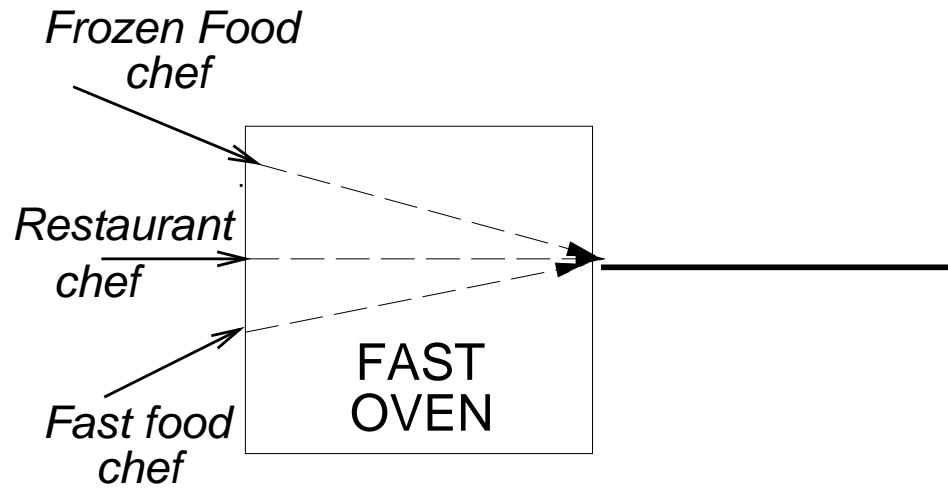


- Crucial for Voice over IP (see Cisco's Specs) to reduce latency. WFQ/DRR etc. only guarantees a fair share of *bandwidth*. MDDR combines priority with DRR.

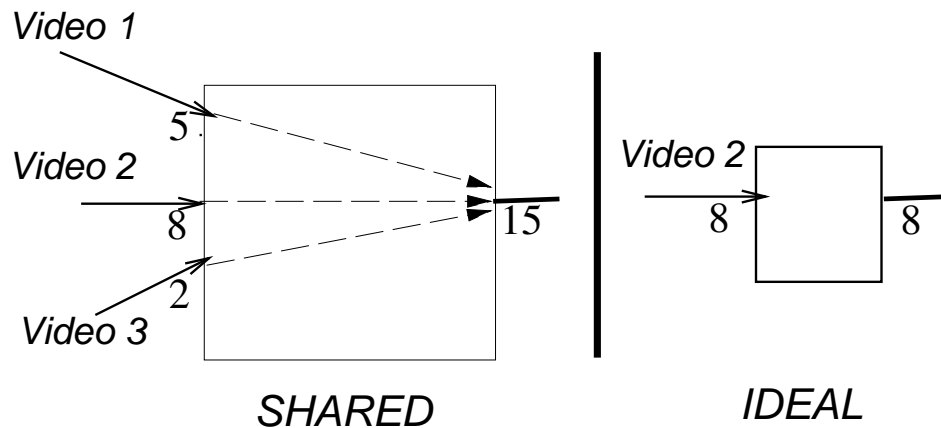
Reservation Protocols

- Reservations are essential for absolute performance guarantees.
- RSVP: works in context of a multicast tree where receivers send requests to senders. Complexity from need to merge reservations for scaling, blockade state, and incremental deployment using Path messages.
- DiffServ revolution replaces RSVP with signalling between Bandwidth Brokers. Not clear whether it will be simpler.

Delay Bounds and Issues

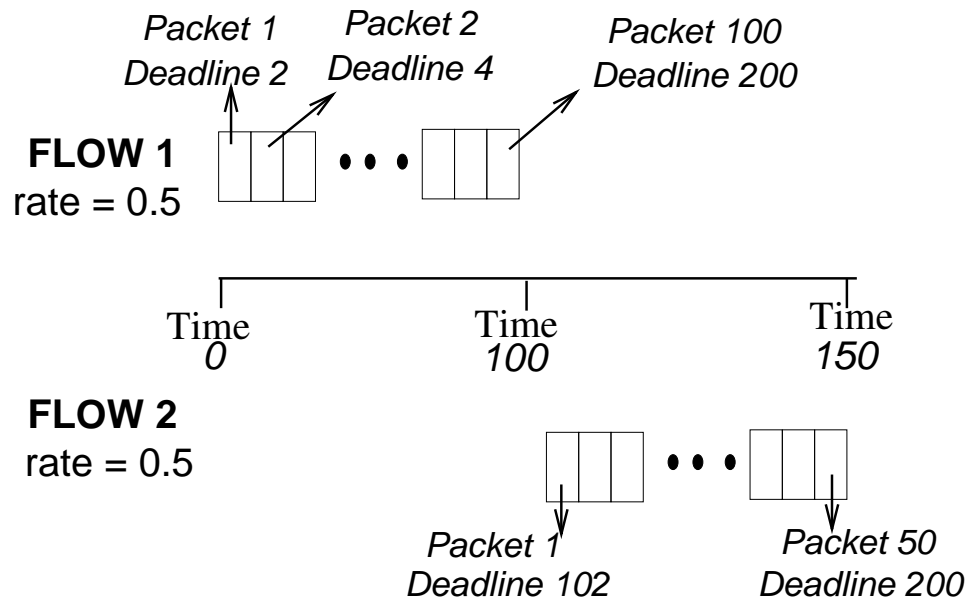


Ideal Delay Bounds and Virtual Clock



- Virtual Clock: Use deadline in ideal system in an earliest deadline first schedule. Can show that if traffic is less than bandwidth of link, all packets meet deadlines.
- Can be done using hardware heaps. Lucent Packet Star.

Some tricky Stuff with Deadlines



- Causes burstiness when there is huge difference in rates.

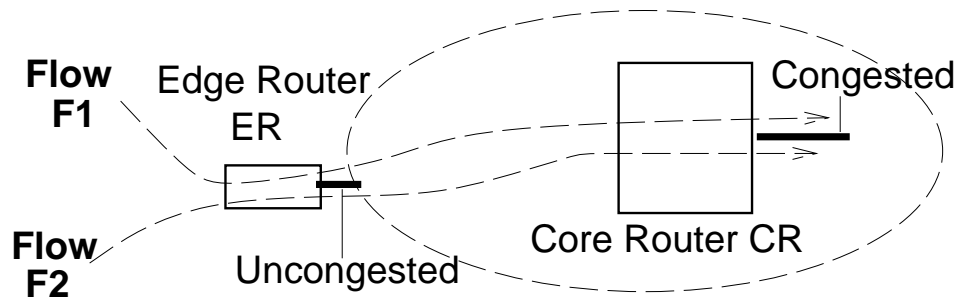
Scaling Problem for Flows

Number of flows in millions. Infeasible to keep per-flow state for QoS. Why not aggregate as in prefixes

- *Random Aggregation:* Stochastic Fair Queuing. Flows hash onto say 1000 buckets and do DRR on 100 buckets. Bad if two bad flows hash onto same bucket.
- *Edge Aggregation:* Diff Serv proposal. Edge routers classify flows into a few aggregation classes that core routers treat differently.
- *Edge Aggregation with Policing:* Prevents unfairness between two flows aggregated into same bandwidth class. Core-stateless.

Core Stateless Fair Queuing

in



RED with attitude.

- Edge router passes a value proportional to degree a flow is oversubscribed.
- Core router does RED based on value *in packet* and not just based on its queue size. So misbehaving flows are dropped with higher probability/

Pirnciples Used

in

<i>NUM</i>	<i>PRINCIPLE</i>	<i>SCHEDULING TECHNIQUE</i>
P7	<i>Use power of two parameters</i>	<i>RED</i>
P3	<i>Use policing not shaping</i>	<i>Token bucket policing</i>
P3 P12 P7	<i>Focus on bandwidth only</i> <i>Maintain list of active queues</i> <i>Use large enough quanta</i>	<i>DRR</i>
P13 P15 P4c P4a	<i>Leap forward not backward</i> <i>Use a heap to sort tags</i> <i>Use a sorting chip</i> <i>Use a d-heap and wide memory</i>	<i>Leap Forward</i> <i>Virtual Clock</i>
P3a	<i>Aggregate by hashing flows</i>	<i>SFQ</i>
P3c P10	<i>Shift work to edge routers</i> <i>Pass class in TOS field</i>	<i>DiffServ</i>
P10	<i>Pass drop probability in header</i>	<i>Core Stateless</i>