University of North Carolina at Chapel Hill

The Evolution of Quality-of-Service on the Internet

Kevin Jeffay
Department of Computer Science

February 2001

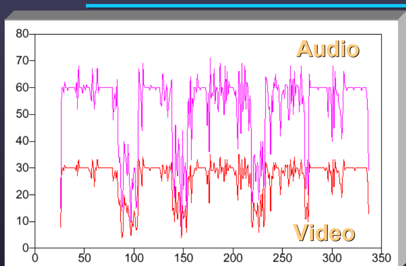
Research Context

Network support for immersive DVEs

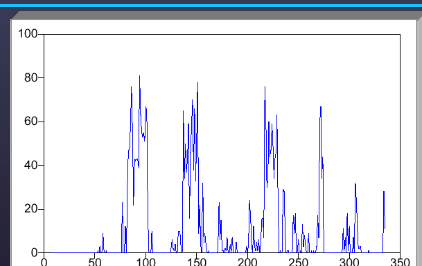
- The Office of the Future
- The nanoManipulator system
- Salient problem characteristics:
  - Continuous media transmission
  - Low latency required for human-to-human communication, and the illusion of immersion

Performance of MM transmission

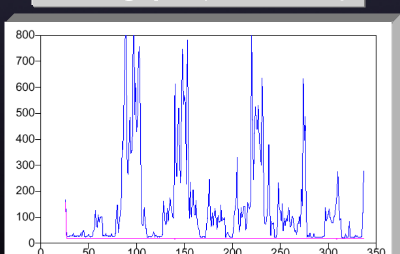
Performance of "raw" transmission



Throughput (frames/sec)



Packet Loss



Audio Latency (ms)

- "Out-of-the-box" ProShare performance
  - Frozen, motionless video
  - Clipped, broken audio

The Evolution of Quality-of-Service on the Internet

Summary

- The Internet is (slowly!) evolving to support quality-of-service
- The current mechanisms for realizing QoS are more about router queue management than virtual circuits
- Virtual circuits were investigated but have been largely abandoned
  - (Did we really need them in the first place?)
- The future Internet will provide router "forwarding behaviors" rather than end-to-end "services"
  - A simple per-hop priority forwarding service suffices

The Evolution of Quality-of-Service on the Internet

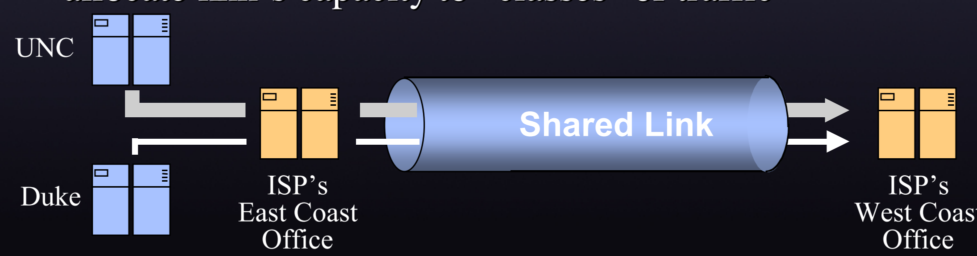
Outline

- The promise of the Internet for real-time communications
- The Integrated Service Architecture for the Internet
  – Reservations, admission control, and scheduling
- The non-deployment of INTSERV
  – What "service" do applications really need?
- The Differentiated Services Architecture for the Internet
  – Active Queue Management for congestion control and quality-of-service

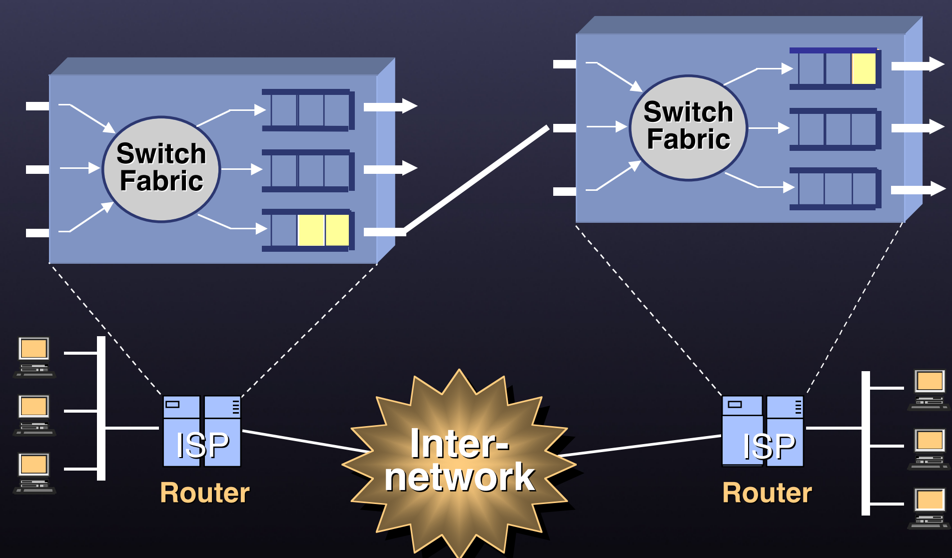
Integrated Services Architecture
Services

An Integrated Services Internet is one that supports:

- "Flows"
  – *real-time communication* — service guarantees
  – *best-effort communication* — today's service model
- Traffic management
  – *controlled link sharing* — enabling a service provider to allocate link's capacity to "classes" of traffic

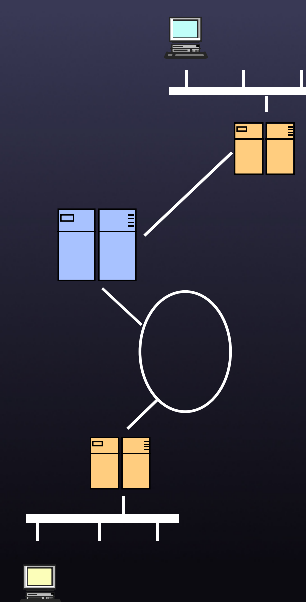


The Nature of Congestion
Queueing delays in routers



Integrated Services Architecture
Axioms

- Resource reservation is required
  – Network elements must maintain per-flow state information and use this information to ensure application performance contracts are met
- Admission control is required
  – To ensure performance contracts are met, network elements must ensure they do not over commit their resources
- Applications must be policed
  – To ensure performance contracts are met, network elements must ensure applications do not claim more resources than they contracted for

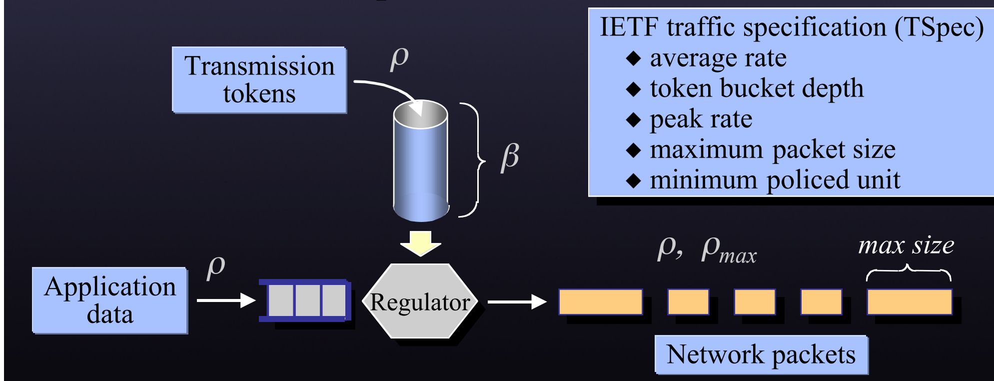


Integrated Services Architecture
Service models for flows

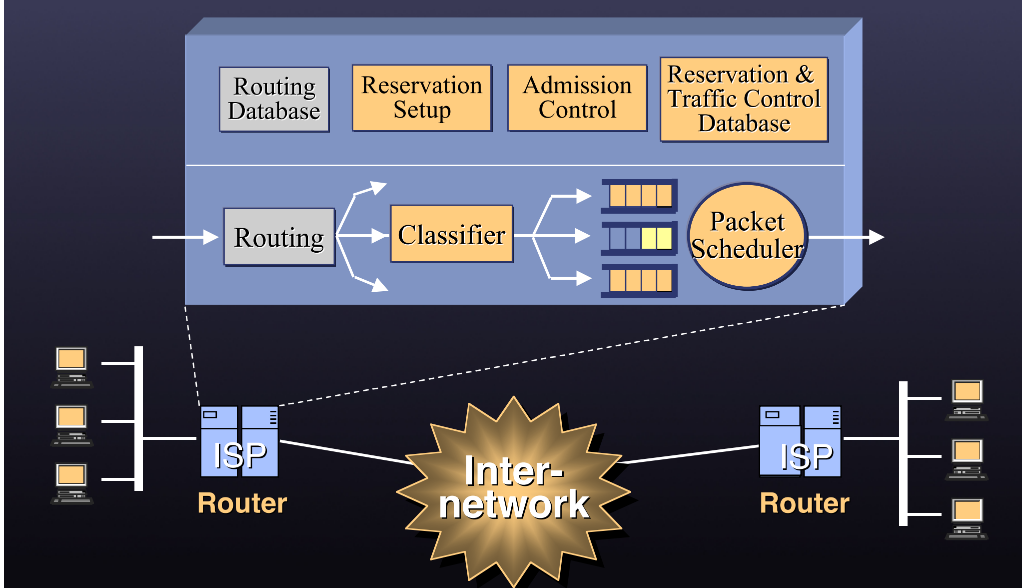
- Integrated services introduces the concept of a *service model*
  - A contract between a sender and the network for a particular quality of service
- Proposed service models
  - *Guaranteed delay* — An application receives a guarantee that all packets will be delivered within a fixed delay bound
  - *Controlled load* — Performance equivalent to that on an "unloaded network"
  - *Best-effort* — Today's service model

Integrated Services Service Models
Flowspecs

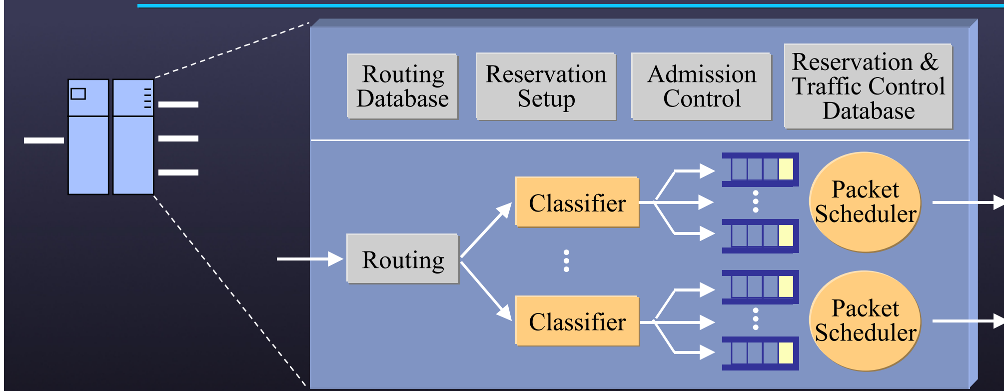
- To receive a service contract an application must specify the service it requires and the traffic it will generate
  - Canonical flow specification — *the token bucket*



Towards QoS Networking
The Integrated Services Architecture

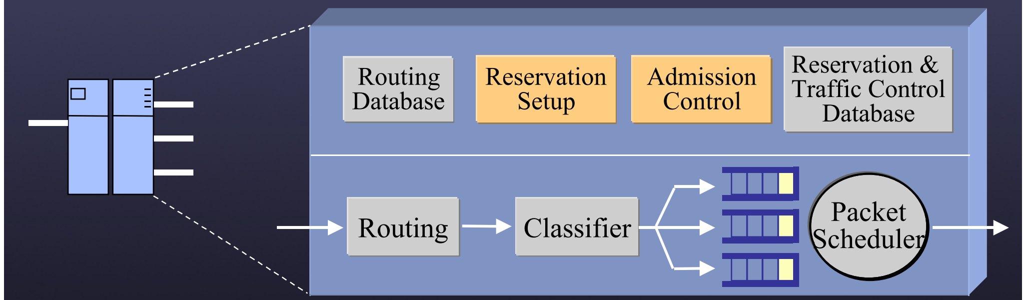


Realizing Integrated Services
Reference implementation components



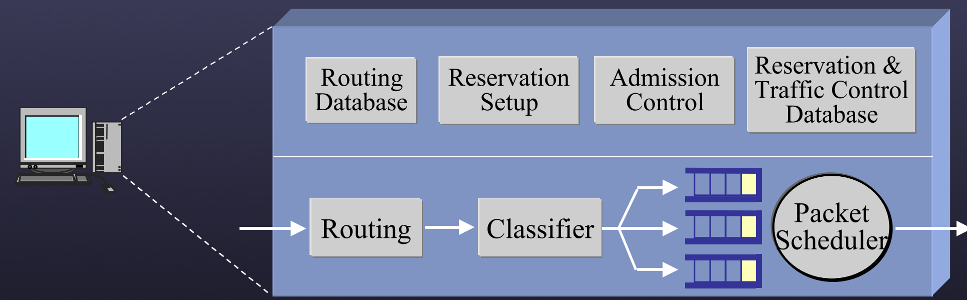
- Classifier — Maps all packets into one or more classes that receive the same service
- Packet Scheduler — Schedules packets for transmission so that performance contracts are enforced

Realizing Integrated Services
Reference implementation components



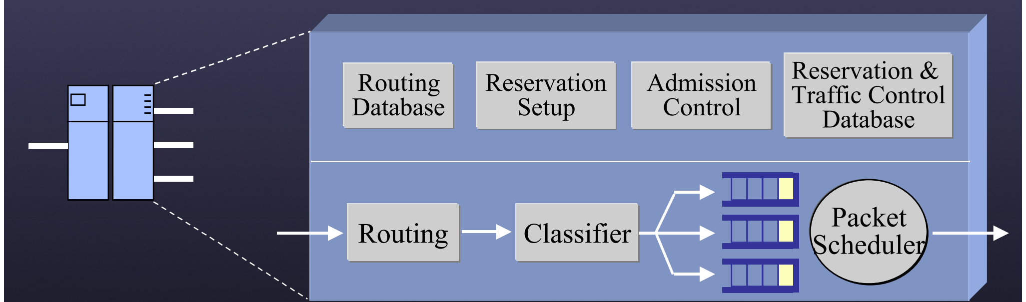
- Reservation setup protocol
  - Mechanism by which flow-specific state is created and maintained
- Admission control procedure
  - The decision procedure that is used to determine if a new flow can be accepted or not

Realizing Integrated Services
Reference implementation components



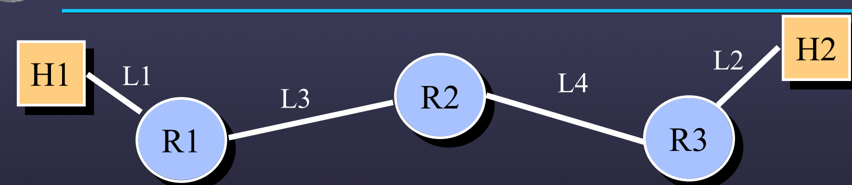
- End systems must support the same logical components
  - A real-time chain is only as strong as its weakest link

Integrated Services Architecture
Architectural components



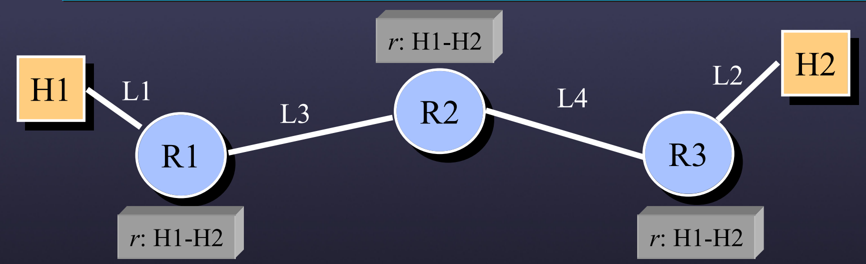
- Flow specifications
- Routing
- Resource reservation
- Admission control
- Packet scheduling

Issues in Resource Reservation
Point-to-point communications



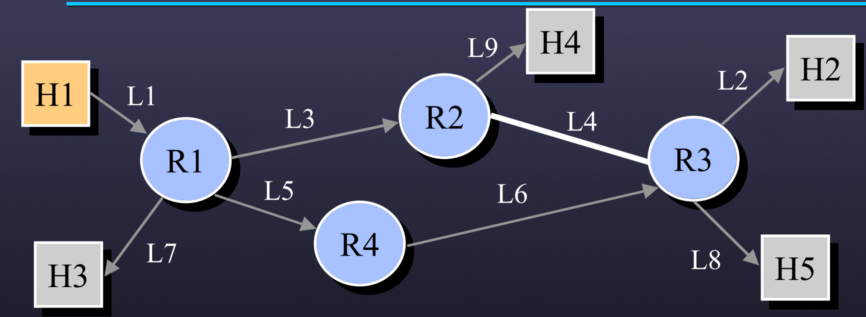
- Goal: Establish a virtual circuit from H1 to H2
  - Reserve "resources" in routers R1, R2, and R3
- Resources are...
  - Link capacity on transmission links
  - Buffer capacity in routers to hold packets in transit
  - CPU capacity at all routers to forward packets from H1 in real-time

Resource Reservation Example
ST-II Two pass reservation protocol



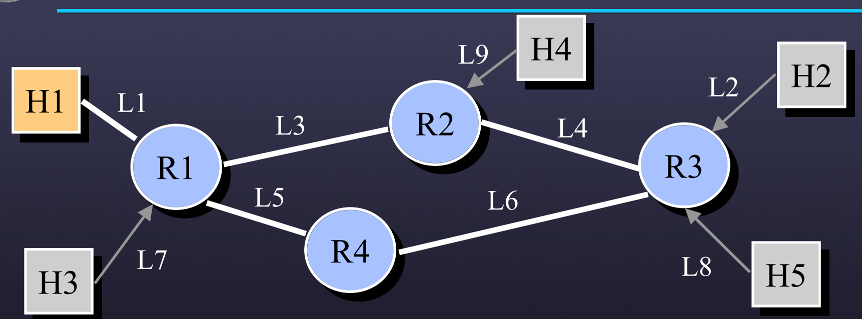
- H1 sends a *connect* message containing a *flowspec* towards H2
  - The connect message is modified as needed by R1-R3
- Upon receipt of the connect, H2 sends an *accept* message back to H1
- Reservations are made when routers receive the accept message

Issues in Resource Reservation
The complexity of supporting multicast



- How do we add/delete new users?
- How do we handle differing link/router capacities?
- How can we avoid over-reserving resources?
- What if the route from H1 to a receiver changes?

RSVP
A receiver initiated reservation protocol



- Receivers initiate reservations
  - Receivers know what bandwidth they want or can handle
  - Places burden of joining/leaving on the involved receiver
  - Admits the possibility of optimizing reservations in routers & switches through aggregation

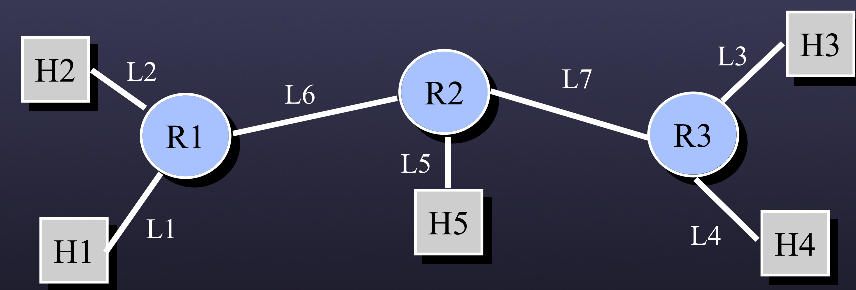
RSVP
Protocol Design

- RSVP is a control protocol
  - RSVP is above IP (like IGMP)
- Reservation is separate from routing
  - Assume only that RSVP and application datagrams are subject to the same routing algorithms
- Reservation and admission control are orthogonal processes
- Reservation state in routers is "soft" and must be periodically refreshed
  - Ensures fault tolerance and allows reservations to be automatically reestablished after route changes

RSVP
Operation Overview

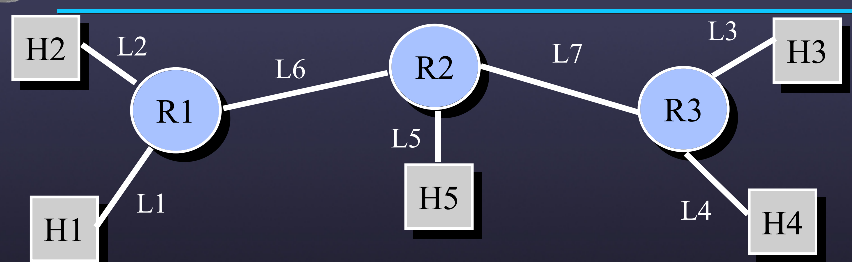
- Senders and receivers join a multicast group
  - (Joining/leaving is performed outside of RSVP)
- Senders advertise their existence
  - Sender to network messages
    - » Path request — make presence of a sender known to network elements
    - » Path teardown — delete path state from routers
- Receivers subscribe to sender data streams
  - Receiver to network messages
    - » Reservation request — reserve resources from sender(s) to receiver
    - » Reservation teardown — delete reservations

RSVP Operation
Example



- 5 hosts, 3 routers/switches
- One multicast group containing all hosts
  - Each host will send and receive media

RSVP Operation Example
Making simple reservations

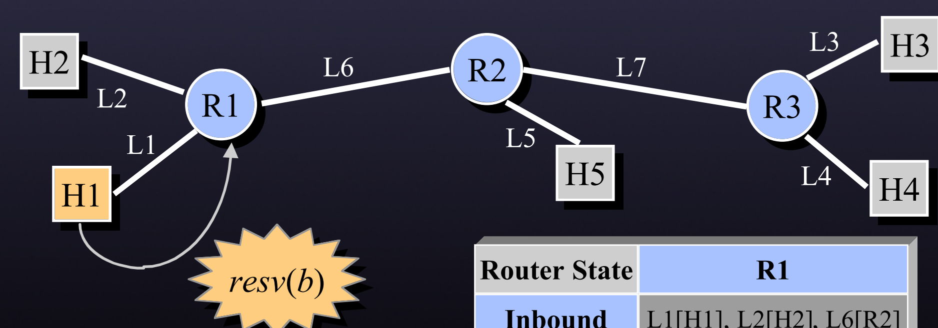


Router State	R1	R2	R3
Inbound	L1[H1], L2[H2], L6[R2]	L5[H5], L6[R1], L7[R3]	L3[H3], L4[H4], L7[R2]
Outbound			

- Assume each router has previously received *path* messages from all sources
- No reservations have been made

Simple Reservation Example
Making a reservation

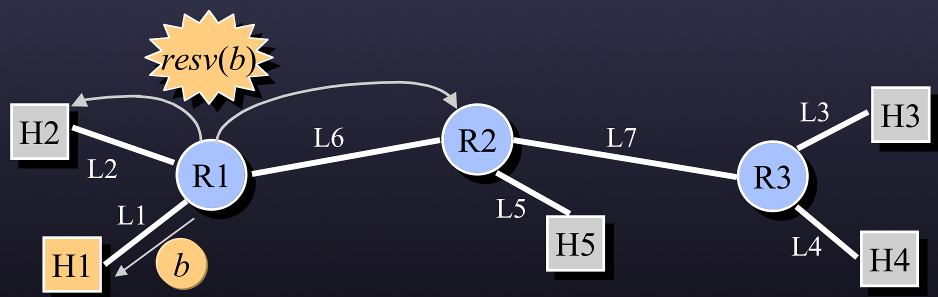
- H1 wants to be able to receive from any sender but only wants 1 stream at a time



Router State	R1
Inbound	L1[H1], L2[H2], L6[R2]
Outbound	

Simple Reservation Example
Forwarding a reservation

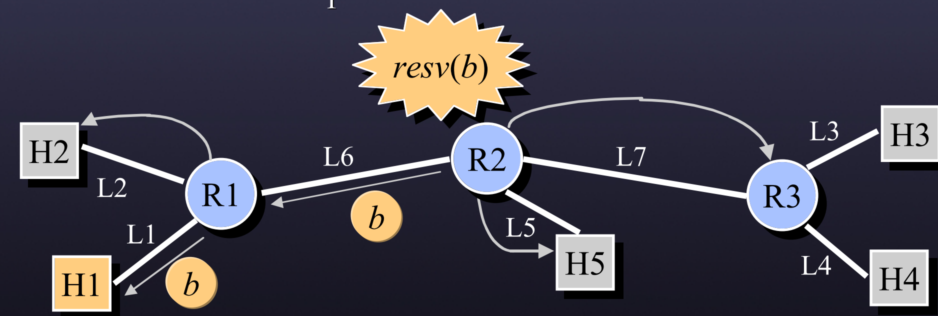
- R1 reserves *b* units of bandwidth *from* R1 *to* H1
- R1 forwards r_1 over all links in its PATH database



Router State	R1	R2	R3
Inbound	L1[H1], L2[H2], L6[R2]	L5[H5], L6[R1], L7[R3]	L3[H3], L4[H4], L7[R2]
Outbound	L1(*b*)		

Simple Reservation Example
Forwarding a reservation

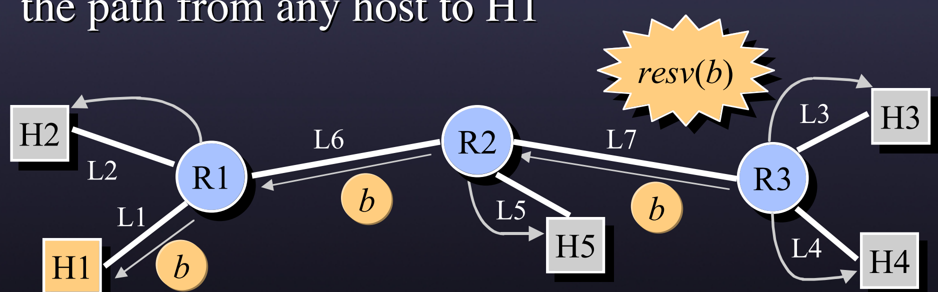
- R2 reserves *b* units of bandwidth *from* R2 *to* R1
- R2 forwards r_1 over L5 & L7



Router State	R1	R2	R3
Inbound	L1[H1], L2[H2], L6[R2]	L5[H5], L6[R1], L7[R3]	L3[H3], L4[H4], L7[R2]
Outbound	L1(*b*)	L6(*b*)	

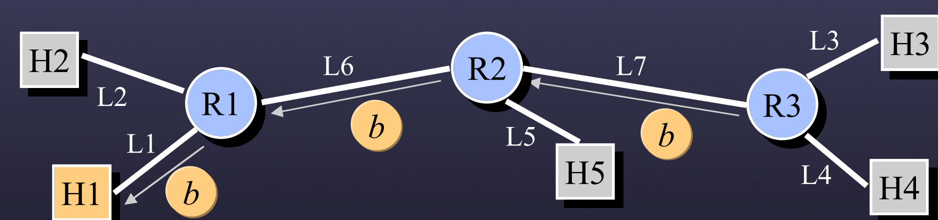
Simple Reservation Example
Forwarding a reservation

- R3 reserves *b* units of bandwidth *from* R3 *to* R2
- Finally, *b* units of bandwidth are reserved along the path from any host to H1



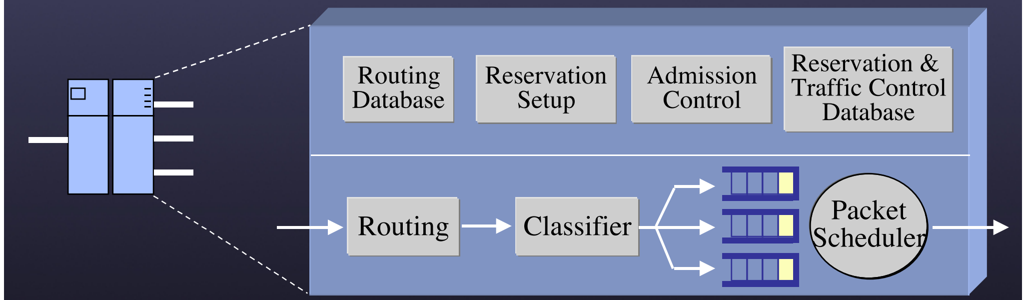
Router State	R1	R2	R3
Inbound	L1[H1], L2[H2], L6[R2]	L5[H5], L6[R1], L7[R3]	L3[H3], L4[H4], L7[R2]
Outbound	L1(*b*)	L6(*b*)	L7(*b*)

Receiver-Initiated Reservations
Summary



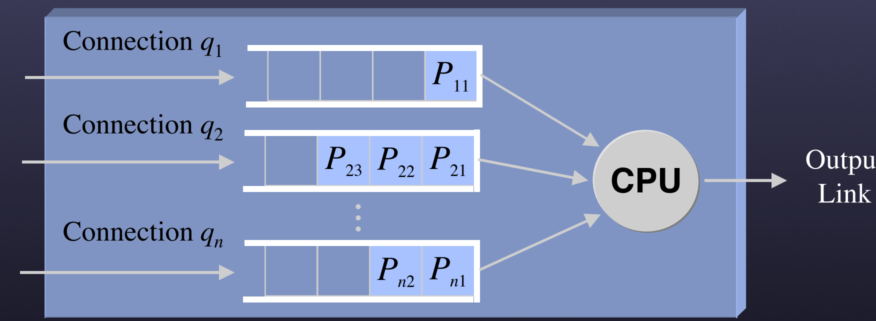
- Resources are not reserved until actually needed
- Reservations are aggregated on intermediate links
- Soft state ensures fault tolerance and provides implicit integration with routing protocols

Integrated Services Architecture
Architectural components



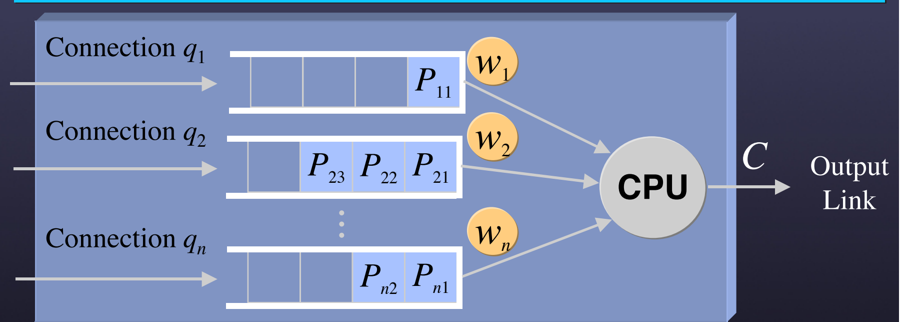
- Flow specifications
- Routing
- Resource reservation
- Admission control
- Packet scheduling

Idealized Resource Allocation
Fluid-flow resource allocation



- Generalized processor sharing (GPS)
  - Service proceeds in *bit-by-bit rounds*
  - 1 bit is serviced from queue i during each round
- Provides *fair* allocation and provides *isolation* from other connections

Fluid-Flow Resource Allocation
Generalized processor scheduling (GPS)

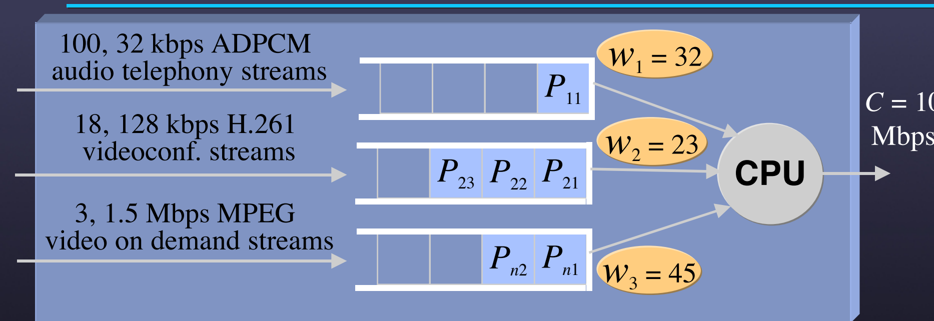


- Connections can be differentiated by integer weights
  - w_i bits transmitted from connection i during each round
  - Each connection receives a *share* of the link's capacity equal to

$$f_i = \frac{w_i}{\sum_j w_j}$$

  - In an interval of length t, connection i transmits $f_i \times t \times C$ bits where C is the link capacity

Generalized Processor Scheduling
Example

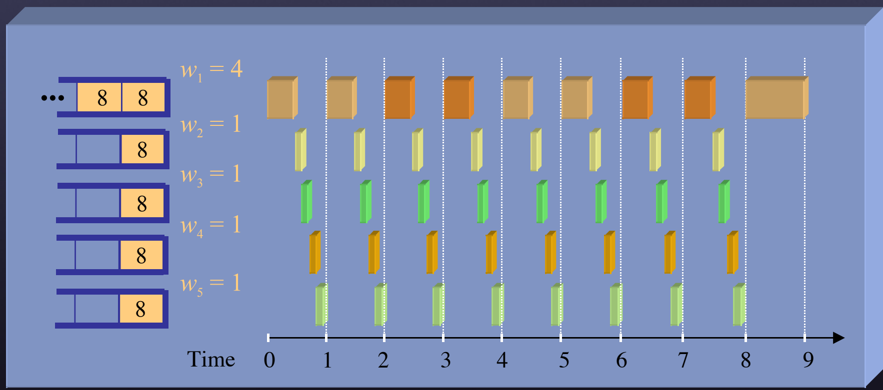


- Consider 3 classes of multimedia connections sharing a 10 Mbps output link in a router
  - Assign each class a weight w so that

$$f_i = \frac{w_i}{\sum_j w_j} \geq \frac{\textit{class bandwidth}}{\textit{link bandwidth}}$$

  - In a round of length 10 μs, connection i transmits w_i bits

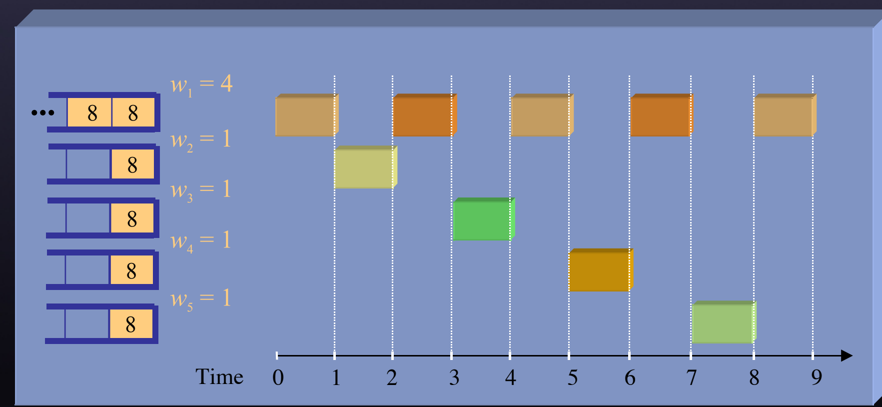
Generalized Processor Scheduling
Bit-by-bit allocation v. packet-by-packet



- In any interval, active connection P_i transmits w_i bits

Generalized Processor Scheduling
Bit-by-bit allocation v. packet-by-packet

- But packet scheduling is inherently non-preemptive!
- Schedule packets by in order of their "finish number"
  - Under "Packet-by-packet GPS" the deviation from true GPS is bounded

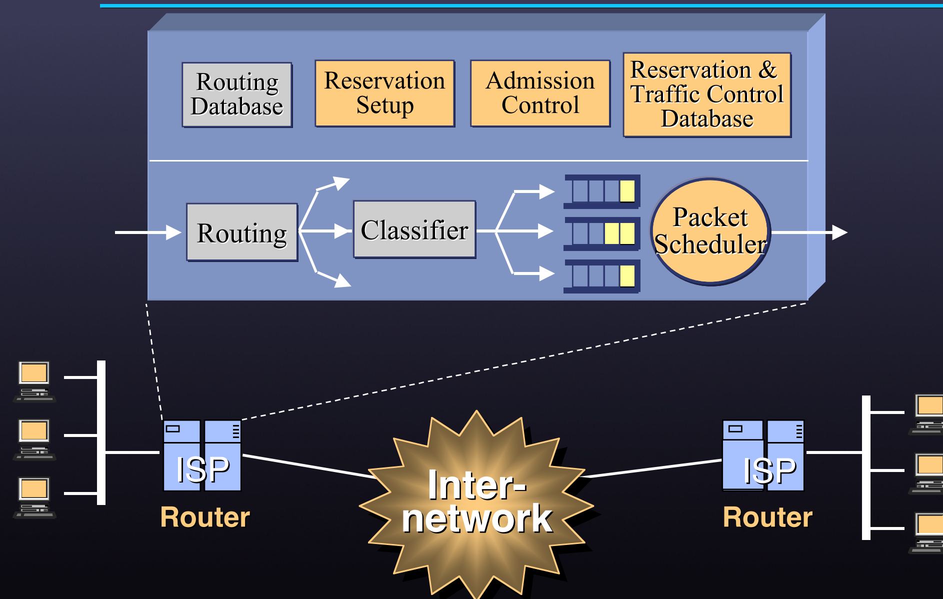


The Evolution of Quality-of-Service on the Internet

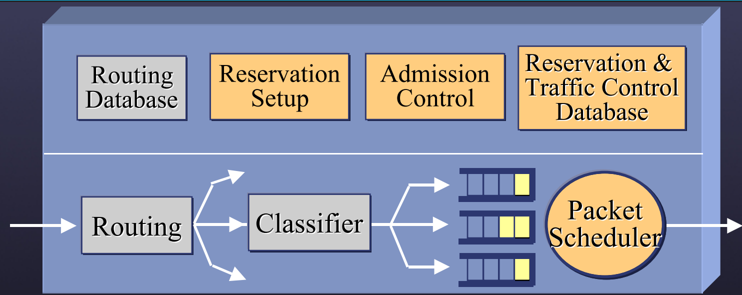
Outline

- The promise of the Internet for real-time communications
- The Integrated Service Architecture for the Internet
  - Reservations, admission control, and scheduling
- The non-deployment of INTSERV
  - What "service" do applications really need?
- The Differentiated Services Architecture for the Internet
  - Active Queue Management for congestion control and quality-of-service

The Integrated Services Architecture
Ahead of its time or fatally flawed?

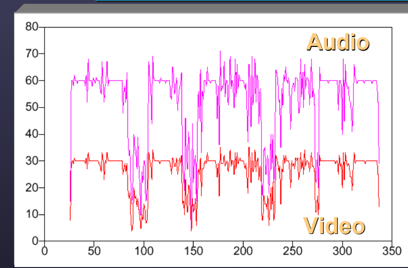


The Integrated Services Architecture
Ahead of its time or fatally flawed?

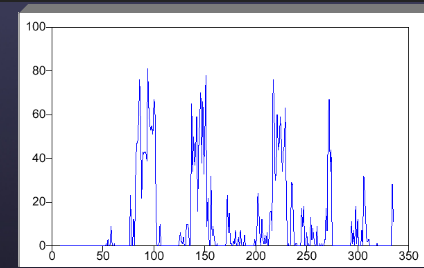


- Guarantees requires per-flow state in every router and switch
  - And guarantees were only modulo route changes
- Algorithmic complexity of reservations and scheduling is non-trivial
- Why would service providers implement this when (arguably) better services can be provided below IP?

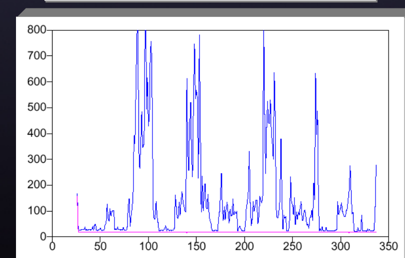
Performance of MM transmission
Performance of "raw" transmission



Throughput (frames/sec)



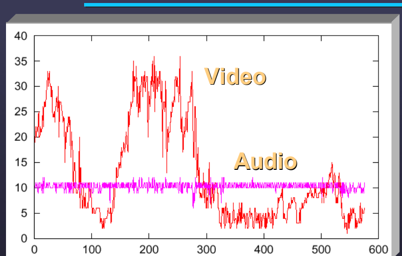
Packet Loss



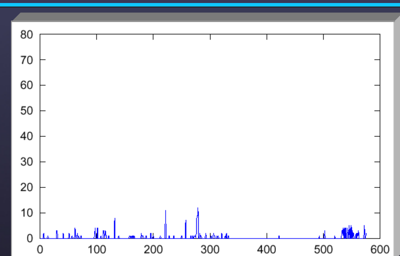
Audio Latency (ms)

- "Out-of-the-box" ProShare performance
  - Frozen, motionless video
  - Clipped, broken audio

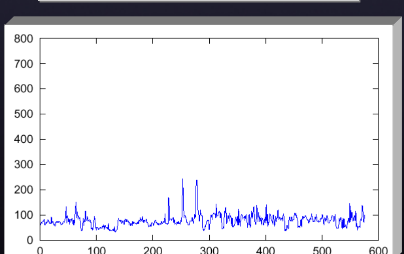
Performance of MM transmission
Performance of adaptive transmission



Throughput (frames/sec)



Packet Loss

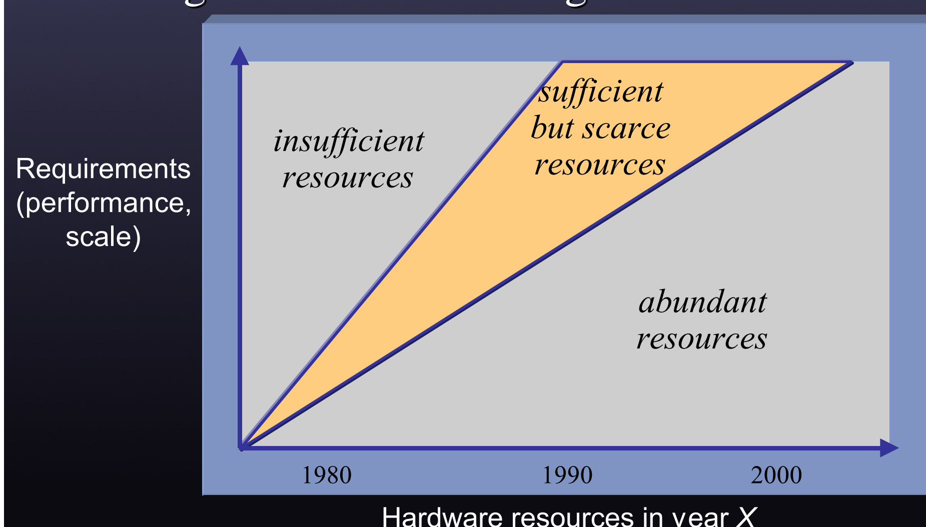


Audio Latency (ms)

- End-system adaptation can ameliorate many of the effects of congestion
  - But can it do so reliably or predictably?
  - (And does it scale?)

Real-time Services on the Internet
The "Window of Scarcity" argument

- Do we need more bandwidth or just better management of the existing bandwidth?

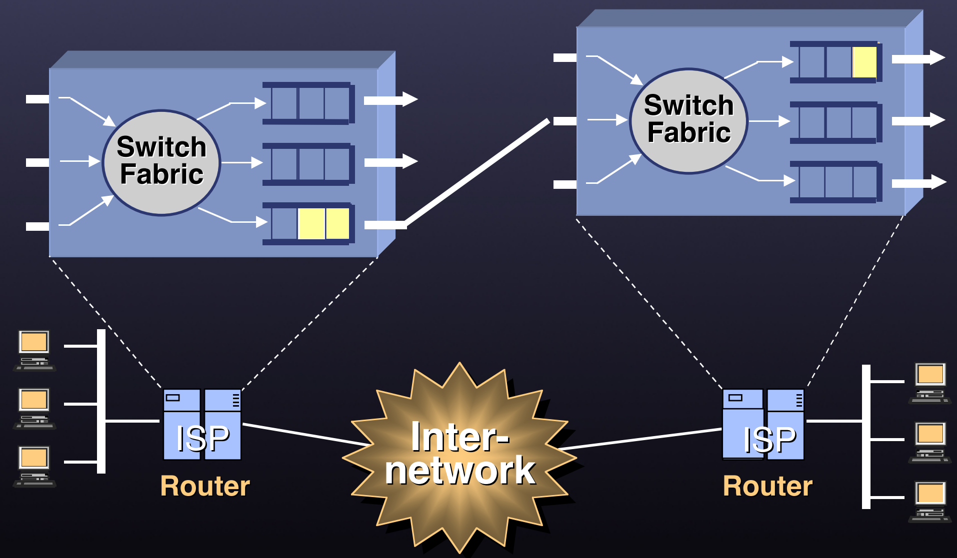


The Evolution of Quality-of-Service on the Internet

Outline

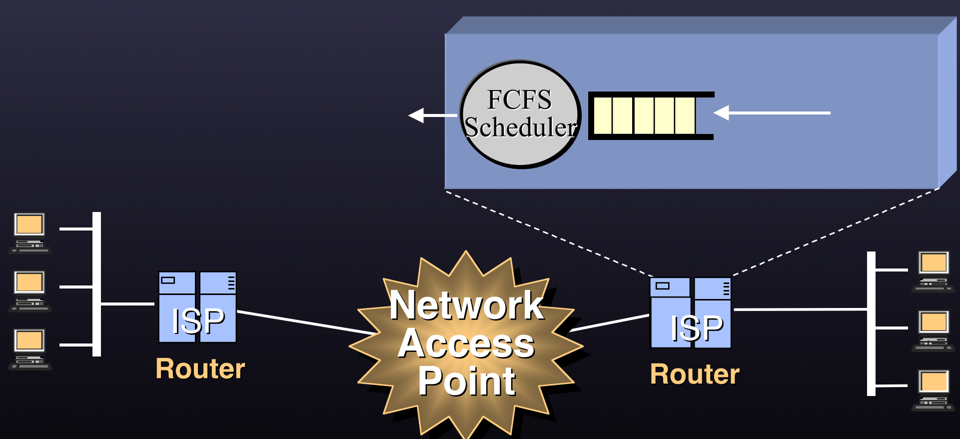
- The promise of the Internet for real-time communications
- The Integrated Service Architecture for the Internet
  - Reservations, admission control, and scheduling
- The non-deployment of INTSERV
  - What “service” do applications really need?
- The Differentiated Services Architecture for the Internet
  - Active Queue Management for congestion control and quality-of-service

The Nature of Congestion
Queueing delays in routers



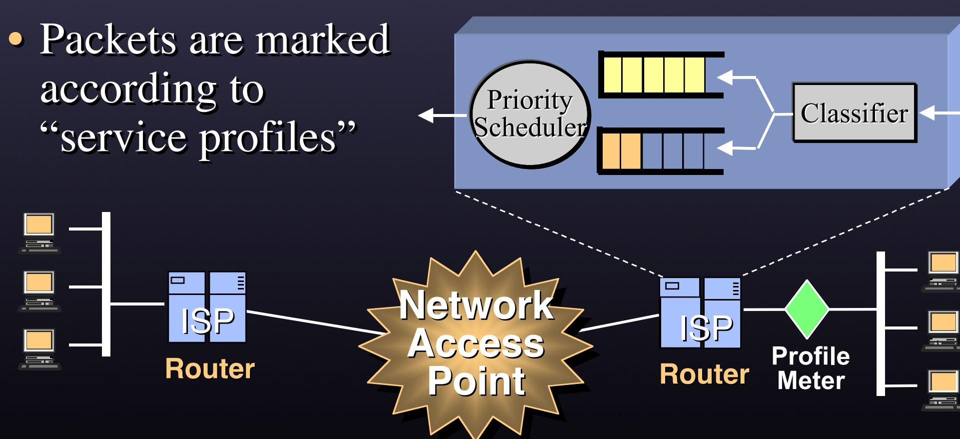
Towards QoS Networking
The differentiated services architecture

- ISPs allocate and sell capacity for a “premium” service



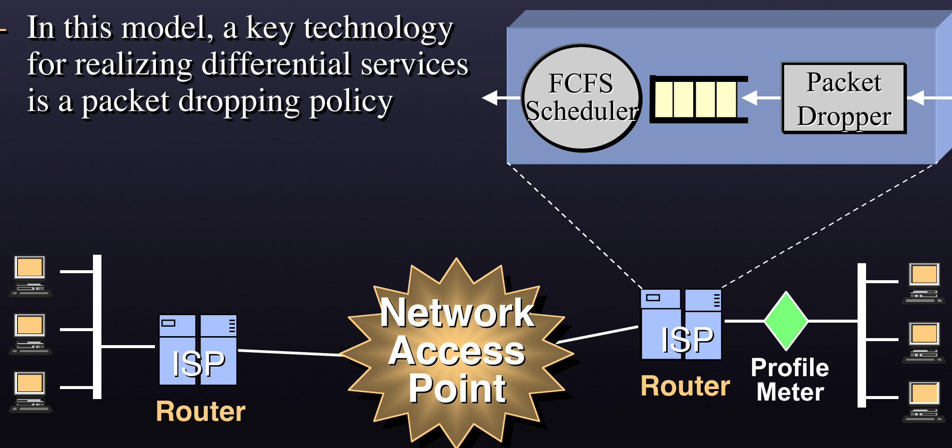
Towards QoS Networking
The differentiated services architecture

- ISPs allocate and sell capacity for a “premium” service
- Packets are marked according to “service profiles”

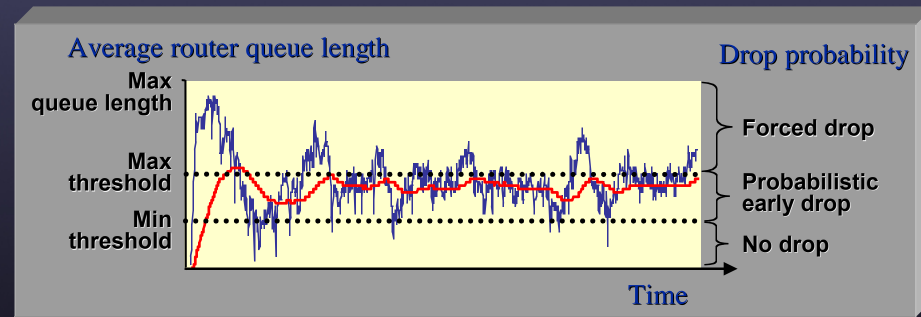


Realizing Differentiated Services
Active queue management

- This is significant utility in realizing differential services with a single router queue
  - In this model, a key technology for realizing differential services is a packet dropping policy

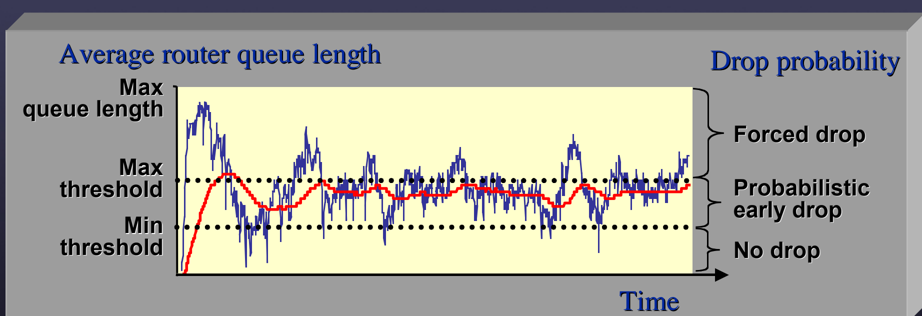


Realizing Differentiated Services
RED active queue management



- Basic mechanism for realizing differentiated services is the *random early detection* (RED) congestion avoidance mechanism

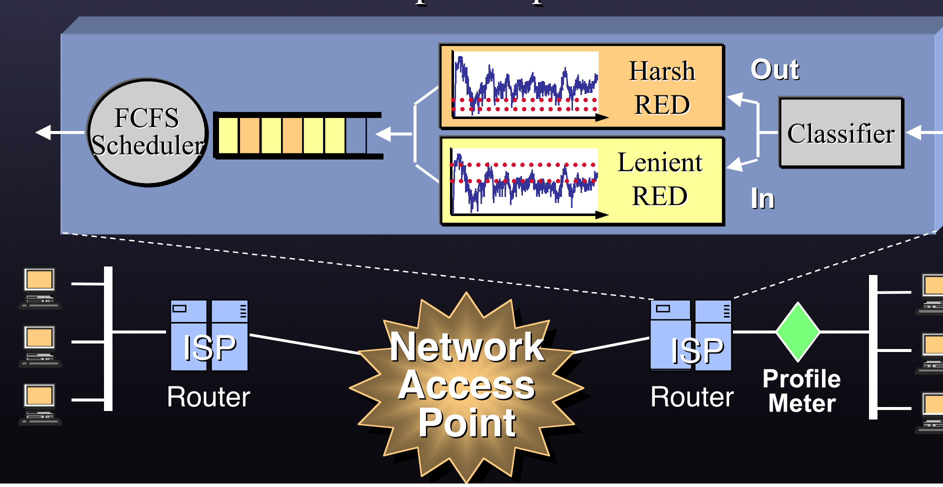
Realizing Differentiated Services
RED active queue management



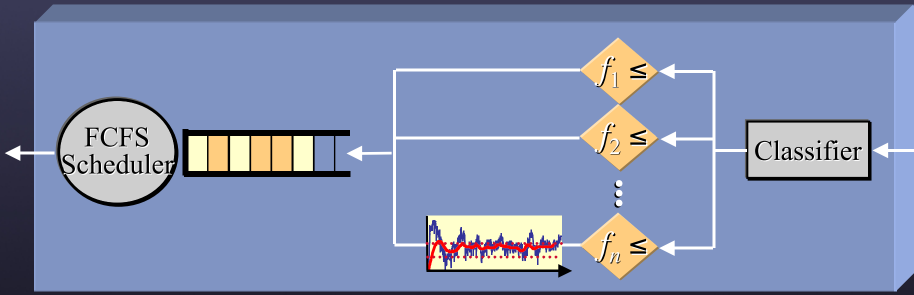
- Random drops avoid lock-out/synchronization effects
  - All flows see the same loss rate
- Early drops avoid full queues
  - Increases effective network utilization ("goodput")
  - Decreases end-to-end latency by decreasing queuing delay

Realizing Differentiated Services
RED & diffserv

- Clark *et al.* RED with "In/Out" (RIO) scheme
  - Apply "harsh RED" to out-of-profile packets and "lenient RED" to in-profile packets



Realizing QoS Through AQM
"Class-based thresholds"



- Designate a set of traffic classes and allocate a fraction of a router's buffer capacity to each class
- Once a class is occupying its (weighted average) limit of queue elements, discard *all* arriving packets
- Within a traffic class, further active queue management may be performed

The Evolution of Quality-of-Service on the Internet

GPS — RIO — CBT — RED — FIFO

Guaranteed QoS Perfect Congestion Control — Better-Than-Best-Effort Forwarding — Advanced Congestion Control — End-System Adaptation to Congestion

- The Internet is evolving to support quality-of-service
  – Capacity allocation & inter-flow protection are required for QoS
- The current mechanisms for realizing QoS are more about router queue management than virtual circuits

The Evolution of Quality-of-Service on the Internet

GPS — RIO — CBT — RED — FIFO

Guaranteed QoS Perfect Congestion Control — Better-Than-Best-Effort Forwarding — Advanced Congestion Control — End-System Adaptation to Congestion

- Active Queue Management can provide performance comparable to packet scheduling...
  – Lower state requirements and algorithmic complexity
- The Internet of tomorrow will provide router "forwarding behaviors" rather than end-to-end "services"

University of North Carolina at Chapel Hill

The Evolution of Quality-of-Service on the Internet

Kevin Jeffay
Department of Computer Science

February 2001