CS 268: Lecture 13

QoS: DiffServ and IntServ

Ion Stoica
Computer Science Division
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley
Berkeley, CA 94720-1776

1

Quality of Service

- Traditional Internet gives single class of best-effort service
  - Even though ToS bits were included in the original IP header
- Treats all packets the same
  - All customers
  - All applications
- Should Internet give better quality service to some packets?
  - Why?
  - Why not?

2

Three Relevant Factors

- Application performance
- Bandwidth required to provide performance
- Complexity/cost of required mechanisms

3

Providing Better Service

- Routing or Forwarding
- Scheduling or Dropping
- Relative or Absolute

4

Relative QoS

- Priority scheduling
  - Favored packets get lower delay and lower drop rate
- Priority dropping
  - All sent packets get same average delay
- Why bother with priority dropping?

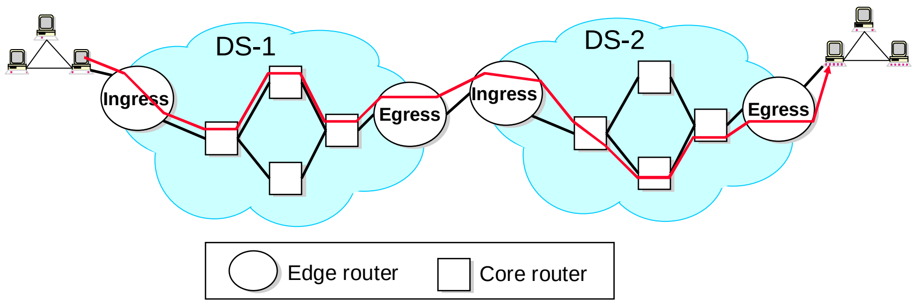
5

Differentiated Services (DiffServ)

- Goal: offer different levels of service
  - Organized around domains
  - Edge and core routers
- Edge routers
  - Sort packets into classes (based on variety of factors)
  - Police/shape traffic
  - Set bits (DSCP) in packet header
- Core routers
  - Handle packet (PHB) based on DSCP

6

DiffServ Architecture



7

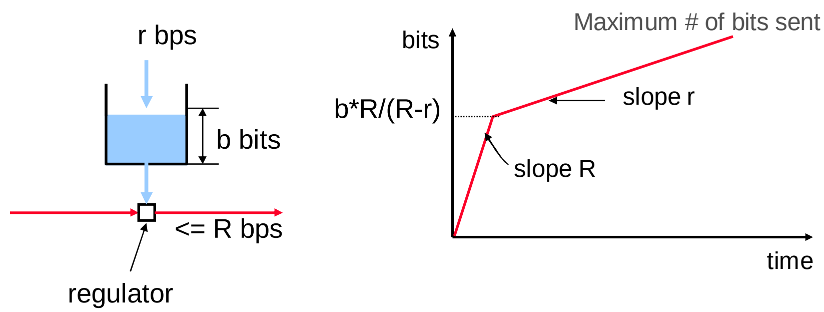
Traffic Policing/Shaping

- Token bucket (r,b)
- Police: if token is available, packet is considered “in”
  - Otherwise considered “out”
- Shape: packet is delayed until token is available

8

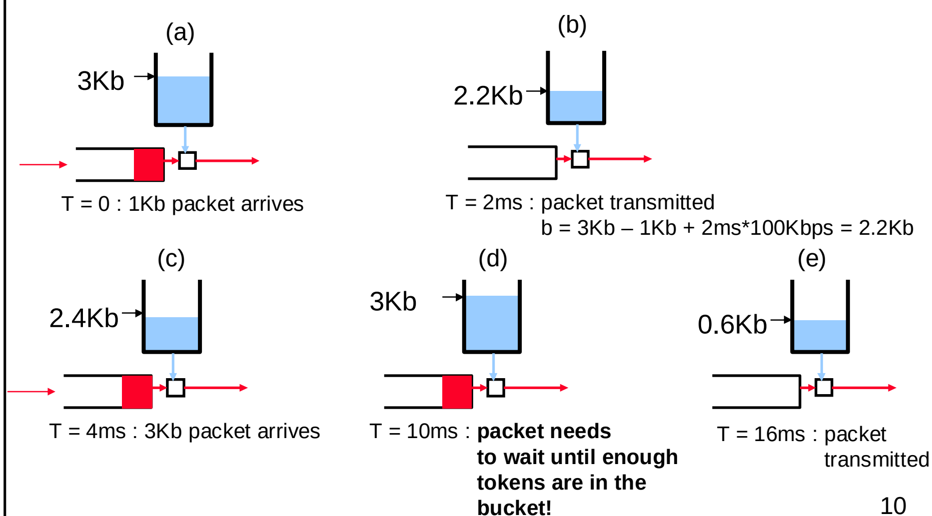
Token Bucket

- Parameters
  - r – average rate, i.e., rate at which tokens fill the bucket
  - b – bucket depth
  - R – maximum link capacity or peak rate (optional parameter)
- A bit is transmitted only when there is an available token



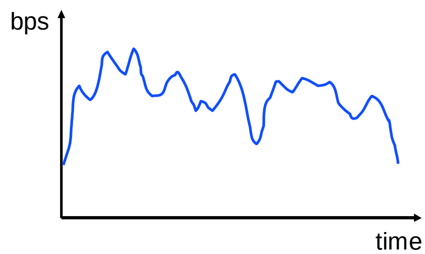
Traffic Enforcement: Example

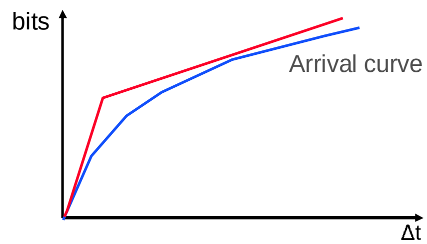
- r = 100 Kbps; b = 3 Kb; R = 500 Kbps



Source Traffic Characterization: Arrival Curve

- Arrival curve – maximum amount of bits transmitted during an interval of time Δt
- Use token bucket to bound the arrival curve

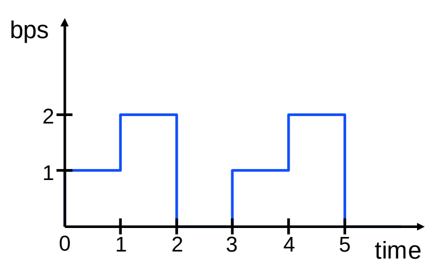


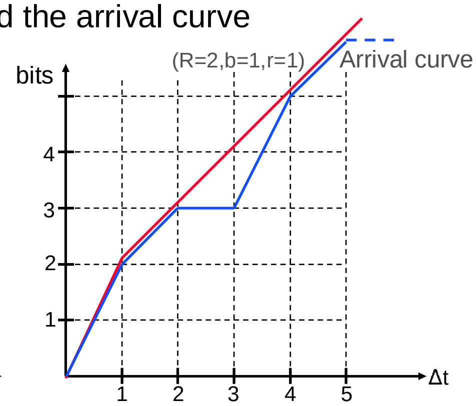


11

Arrival Curve: Example

- Arrival curve – maximum amount of bits transmitted during an interval of time Δt
- Use token bucket to bound the arrival curve

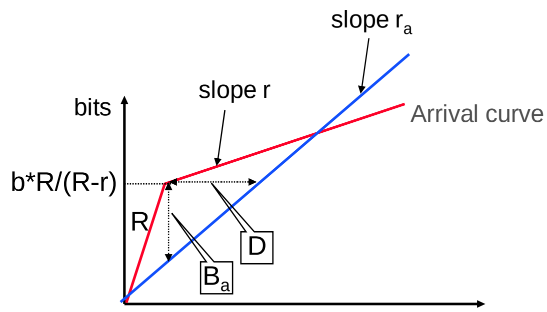




12

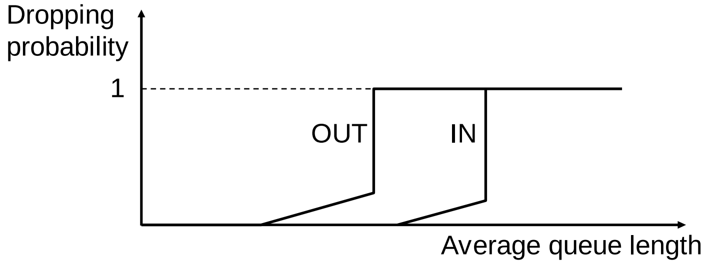
QoS Guarantees: Per-hop Reservation

- End-host: specify
  - the arrival rate characterized by token-bucket with parameters (b,r,R)
  - the maximum maximum admissible delay D, no losses
- Router: allocate bandwidth r_a and buffer space B_a such that
  - no packet is dropped
  - no packet experiences a delay larger than D



Implementing Drop Priority

- RED in/out (RIO)
- Separate dropping curves for in and out traffic
  - Out curve measures all packets
  - In curve measures only in packets



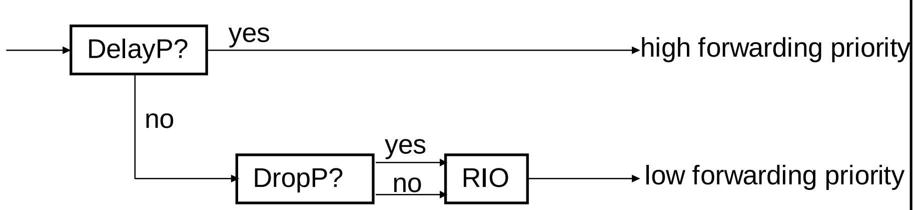
Sender and Receiver Versions

- Sender-based version:
  - Sender (or token bucket next to sender) sets in/out bits
  - Routers service with priority
- Receiver-based version: use ECN
  - Put incoming packets through token bucket
  - If packet is “in”, cancel any ECN bits
  - Receiver only told about congestion for “out” packets

15

Combining Drop and Delay Priority

- Delay priority traffic gets high forwarding priority
- Drop priority traffic uses RIO



16

Why Does Giving Priority Help?

- Making service for one class of traffic better means that service for another class of traffic must get worse
- Why does that help?

17

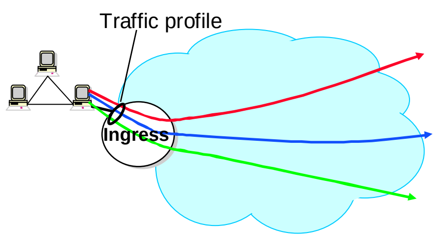
From Relative to Absolute Service

- Priority mechanisms can only deliver absolute assurances if total load is regulated
- Service Level Agreements (SLAs) specify:
  - Amount user (organization, etc.) can send
  - Level of service delivered to that traffic
- Premium Service (DiffServ) offers low (unspecified) delay and no drops
  - Acceptance of proposed SLAs managed by "Bandwidth Broker"
  - Only over long time scales

18

Providing Assurances

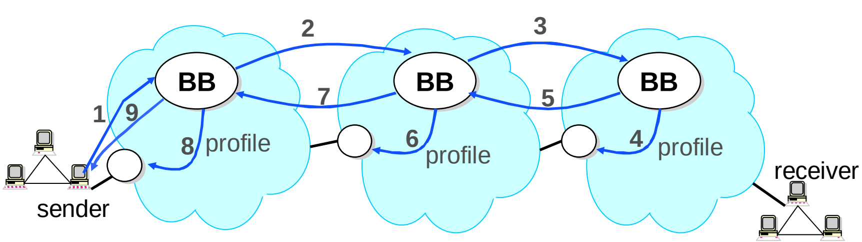
- SLAs are typically defined without restriction on destination
- Can't provision network efficiently, but may not matter



19

Inter-Domain Premium DiffServ

- Achieve end-to-end bandwidth guarantee
- But is this done for all paths?



20

From DiffServ to IntServ

- Can easily provide some traffic better service than others
  - Making absolute assurances requires controlling load
- DiffServ worst-case provisioning very inefficient
  - Based on aggregate offered load, not for a specific path
- What about fine-grain assurances about QoS?
  - Per-flow, not per traffic class
- Requires admission control for each flow
  - E.g., reservations

21

Major Philosophical Change

- Per-flow admission control is drastic change to the Internet
  - But best-effort still available (used for most traffic)
- We will first discuss whether this is a good idea
  - Going back to basics about application performance, etc.
- We will then talk about how one might do this
  - Cursory overview, because details are in the dustbin of history

22

Reservations or Best-Effort

- Basic question:
  - Should we admit all flows (BE), or
  - Refuse some to preserve good service for current flows (R)
- Precedents:
  - The telephone network uses admission control
  - The current Internet does not
- Which one is right? Huge ideological battle!!
- How can we decide?
  - Which provides better application performance?

23

Modeling Application Performance

- Not a simple function of delay/jitter/loss
- Depends on user perception
  - e.g., picture quality, etc.
- Depends on adaptive application behavior
  - Adjust sending rate
  - Adjust coding (to mask errors)
  - Adjust “playback point” (later)
- For a given application, can describe performance as a function of available bandwidth

24

Classes of Application

- Traditional data applications: “elastic”
  - Tolerant of delay
  - Tolerant of loss
- Streaming media applications: “real-time”
  - Less tolerant of delay
  - Less tolerant of loss
  - Often of the “playback” variety

25

Playback Applications

- Video/audio stream being sent
- “Played back” at receiver
- Receiver picks time to play back content
  - “playback point”
- Playback point:
  - Moves: distortion
  - Late: delay
  - Misses packets: “drops”

26

The Overprovisioning Debate

- Some claim bandwidth is plentiful everywhere
  - Cheap
  - Or needed for fail-over
- But that's within core of ISPs
- Bandwidth is scarce:
  - At edge
  - Between providers
- Intserv would help pay for bandwidth in those places

27

IntServ

- IntServ = Integrated Services Internet
- Goal: support wider variety of services in single architecture
- Effort largely led by PARC, MIT, USC/ISI

28

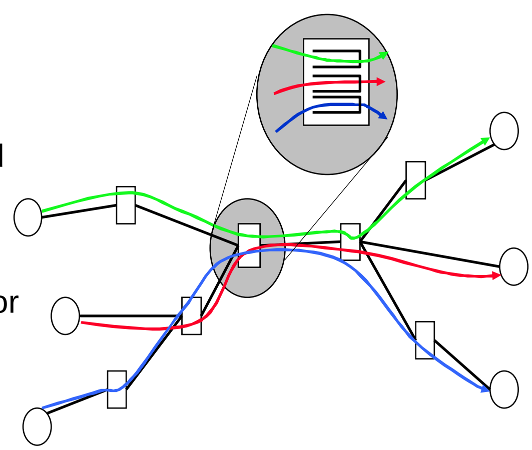
Key IntServ Design Decisions

- Reservations are made by endpoints
  - Network is not making guesses about application requirements
- IntServ is multicast-oriented
  - Assumed that large broadcasts would be a driver of both IntServ and multicast
  - Reservations made by receivers
- Soft-state: state in routers always refreshed by endpoints
- Service guarantees are end-to-end on a per-flow basis

29

Integrated Services Internet

- Flow is QoS abstraction
- Each flow has a fixed or stable path
- Routers along the path maintain state for the flow
- State is used to deliver appropriate service



30

IntServ Mechanisms

- Reservation protocol: transmits service request to network
  - TSpec: traffic description
  - RSpec: service description
- Admission control: determines whether to accept request
- Packet scheduling: ensures router meets service rqmts
- Routing: pin routes, look for resource-rich routes

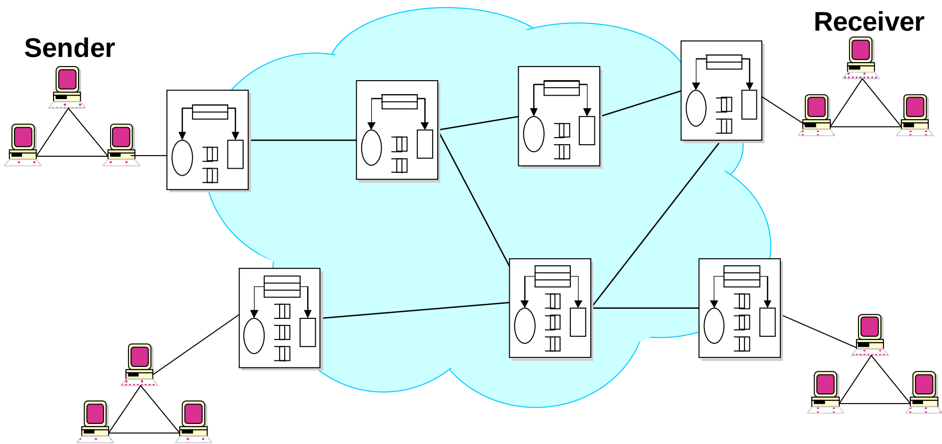
31

IntServ Services

- Kinds of service assurances:
  - Guaranteed (never fails unless major failure)
  - Predictive (will almost never fail)
- Corresponding admission control:
  - Guaranteed: worst-case
    - No guessing about traffic
  - Predictive: measurement-based
    - Gamble on aggregate behavior changing slowly

32

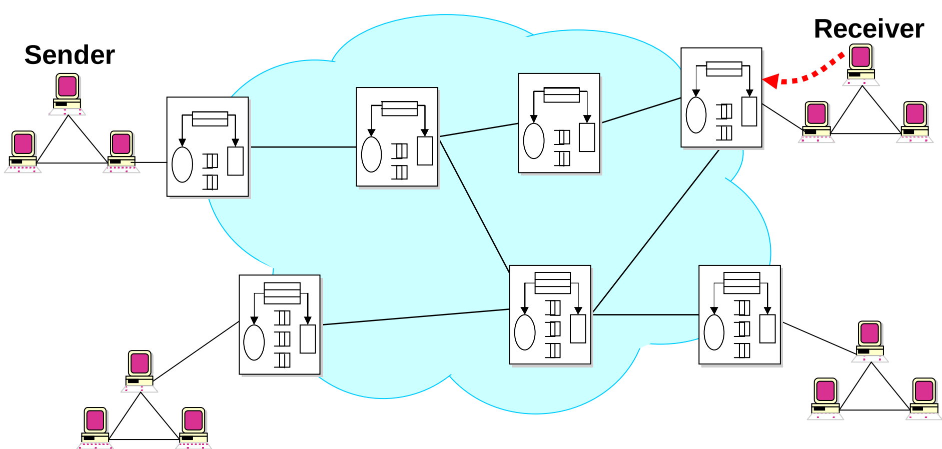
Integrated Services Example



33

Integrated Services Example

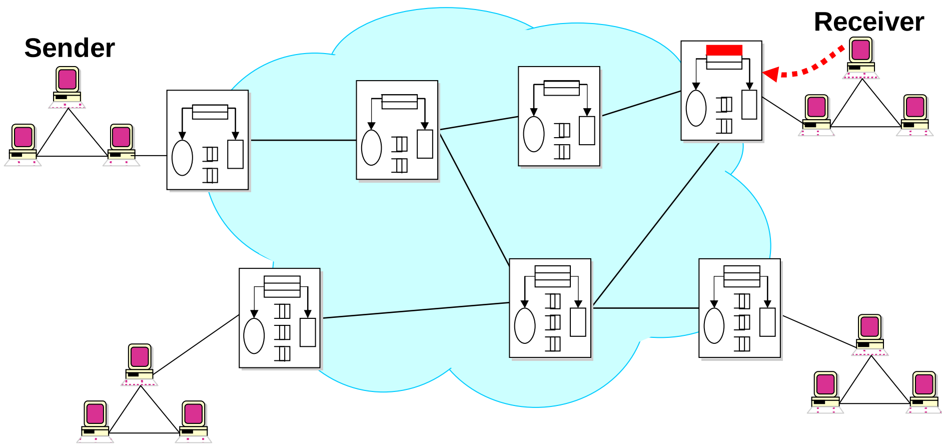
- Allocate resources - perform per-flow admission control



34

Integrated Services Example

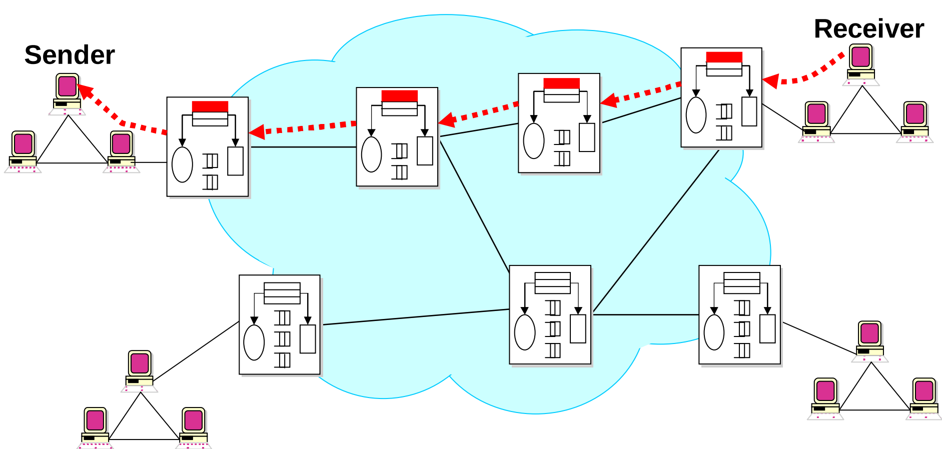
- Install per-flow state



35

Integrated Services Example

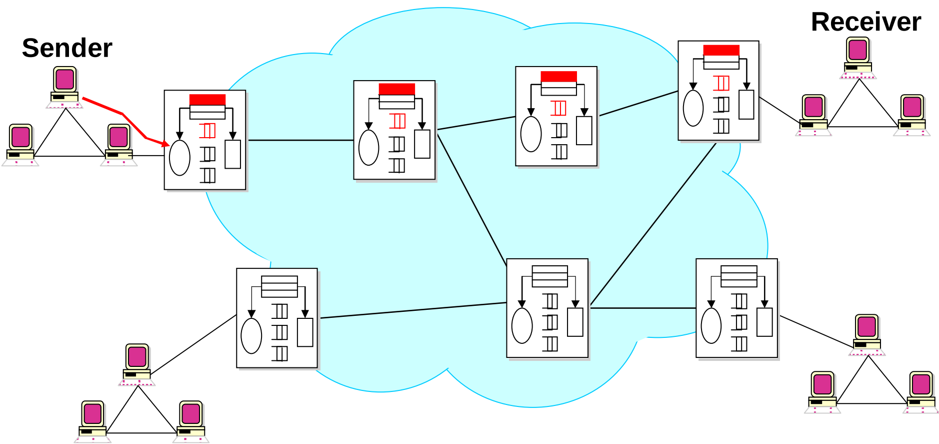
- Install per flow state



36

Integrated Services Example: Data Path

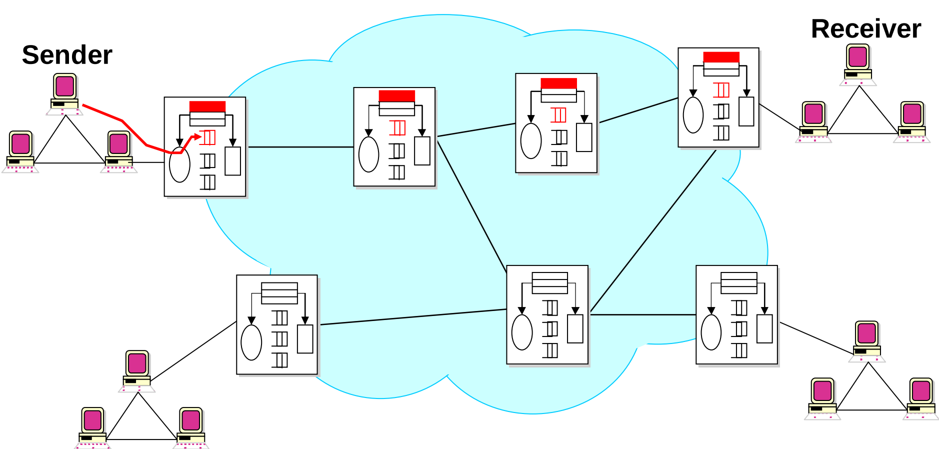
- Per-flow classification



37

Integrated Services Example: Data Path

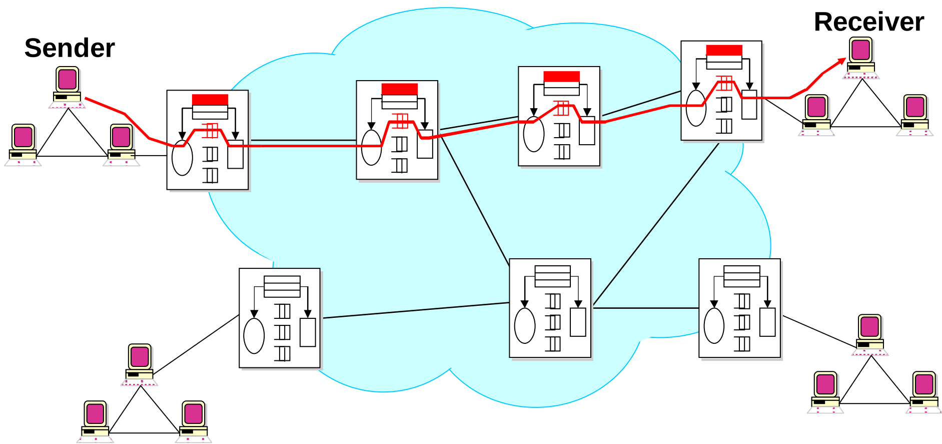
- Per-flow buffer management



38

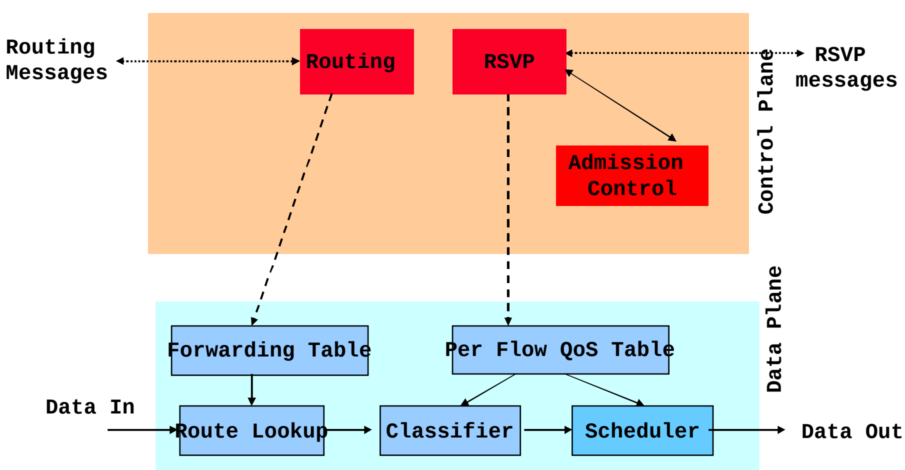
Integrated Services Example

- Per-flow scheduling



39

How Things Fit Together



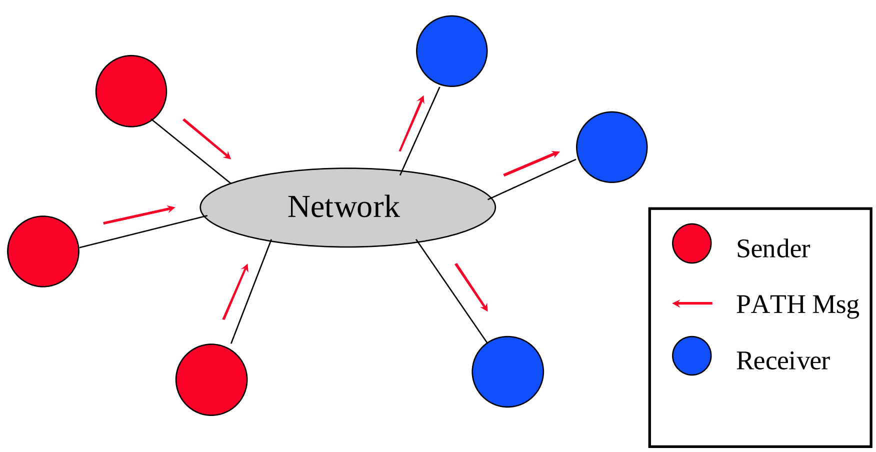
40

RSVP Reservation Protocol

- Performs signaling to set up reservation state for a session
- A session is a simplex data flow sent to a unicast or a multicast address, characterized by
  - <IP dest, protocol number, port number>
- Multiple senders and receivers can be in same session

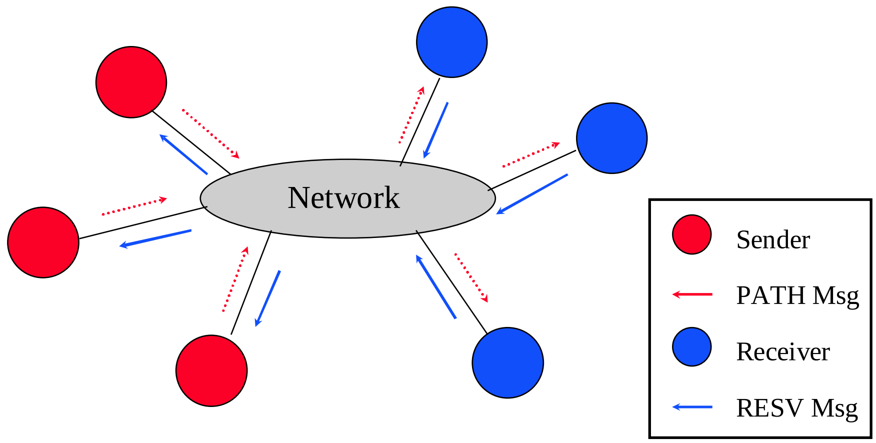
41

The Big Picture



42

The Big Picture (2)



43

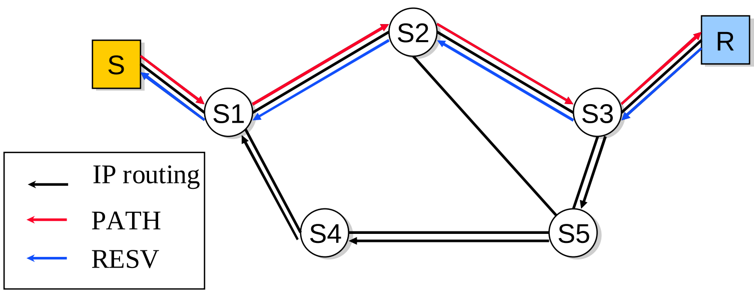
RSVP Basic Operations

- Sender: sends PATH message via the data delivery path
  - Set up the path state each router including the address of previous hop
- Receiver sends RESV message on the reverse path
  - Specifies the reservation style, QoS desired (RSpec)
  - Set up the reservation state at each router
- Things to notice
  - Receiver initiated reservation
  - Decouple routing from reservation

44

Route Pinning

- Problem: asymmetric routes
  - You may reserve resources on R→S3→S5→S4→S1→S, but data travels on S→S1→S2→S3→R !
- Solution: use PATH to remember direct path from S to R, i.e., perform route pinning



45

PATH and RESV messages

- PATH also specifies
  - Source traffic characteristics
    - Use token bucket
- RESV specifies
  - Service requirements
  - Source traffic characteristics (from PATH)
  - Filter specification, i.e., what senders can use reservation
  - Based on these routers perform reservation

46

Reservation Style

- Motivation: achieve more efficient resource
- Observation: in a video conferencing when there are M senders, only a few are active simultaneously
  - Multiple senders can share the same reservation
- Various reservation styles specify different rules for sharing among senders
- Key distinction:
  - Reserved resources (bandwidth)
  - Which packets use those resources

47

Reservation Styles: Filters

- Wildcard filter: all session packets share resources
  - Good for small number of simultaneously active senders
- Fixed filter: no sharing among senders, sender explicitly identified in reservation
  - Sources cannot be modified over time
  - Allows reserved resources to be targeted to particular paths
- Dynamic filter: resource shared by senders that are (explicitly) specified
  - Sources can be modified over time
  - Switching between speakers at a conference

48

What Did We Miss?

- Make aggregation central to design
  - In core, don't want to keep track of each flow
  - Don't want to process each RESV message
- Economics: user/provider and provider/provider
  - We talked about it (at great length) but didn't realize how inflexible the providers would be
- Too complicated: filter styles a waste of time
- Multicast focus?

49