Network QoS
371-2-0213

Lecture 2

Gabriel Scalosub

Outline

1. IntServ
   - Key Components
   - Guaranteed Service
   - Controlled Load Service
2. RSVP Protocol
   - Basic Features
   - RSVP Messages
   - Reservation Styles

IntServ - Key Components

- Main motivation: Multimedia (e.g., video conferencing)
  - multicast
- Main concern: *worst-case delay*
- Main mechanism: BW reservation
- Key components:
  - per-flow reservation-based
    - installs state
    - robust to topology change
    - a^3: authorization, authentication, accounting
  - reservation is independent of route selection
    - route selection is crucial
    - often NP-hard (especially with multiple objectives)
    - addressed via traffic engineering (and MPLS)
  - admission control:
    - accept/reject reservation
    - monitor/measure available resources
    - parameter based ("determ.") vs. measurement based ("prob.")

Control Plane

IntServ - Key Components

- Main motivation: Multimedia (e.g., video conferencing)
  - multicast
- Main concern: *worst-case delay*
- Main mechanism: BW reservation
- Key components:
  - flow identification (5-tuple)
    - source IP
    - destination IP
    - protocol ID
    - source port
    - destination port
  - scheduling
    - FIFO
    - priority scheduling
    - weighted fair queuing (WFQ)

Data Plane

Admission Control Metrics

- Common metrics for admission control
  - Simple sum
    - based on reservations: $L + r \leq C$
  - Measured sum
    - based on measured load: $\overline{L} + r \leq \delta C$
    - δ: target utilization (allowing estimation error)
  - Acceptance region
    - requires statistical traffic model
    - maximizes utilization against loss
  - Equivalent bandwidth $C(p)$
    - requires statistical traffic model
    - bandwidth requirement of all flows exceeds $C(p)$ with probability $\leq p$.

r:	new flow rate
L:	committed load
$\overline{L}$:	estimated load
C:	capacity

Load Estimation

- Approaches to load estimation (E_t: estimate at time t)
  - Exponential averaging:
    - Let M_t be the measurement at time t.
    - Updating estimate:

$$E_{t+1} = (1 - \alpha)E_t + \alpha M_t$$

    - α is a smoothing factor (small: smooth, large: fast-adapting)
  - Time-window estimation:
    - Series of intervals I_j, $j = 1, 2 \ldots$
    - C_j is the average load during I_j
    - For any I_j ending at time t,

$$E_t = \max\{C_{j-n}, C_{j-n+1}, \ldots, C_j\}$$

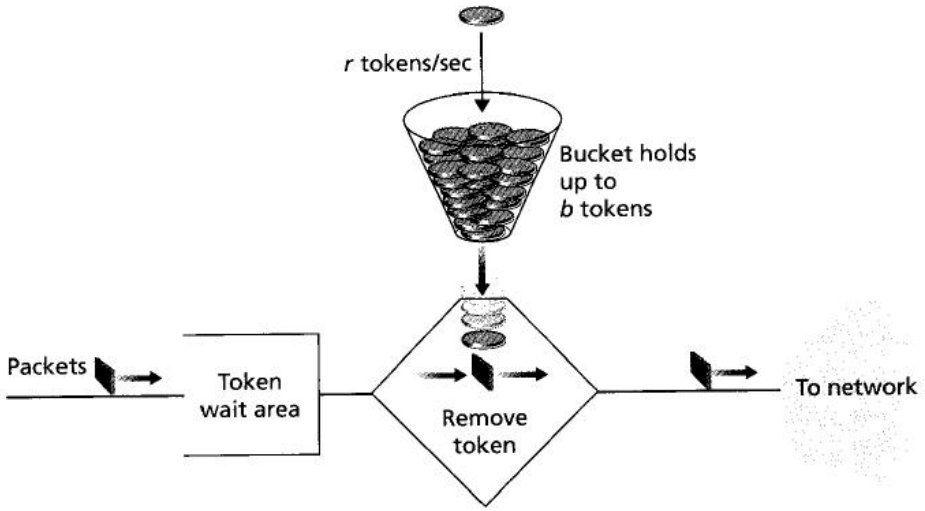
    - smoothing: depends on amount of overlap between consecutive intervals

IntServ Services

- Recall IntServ provides two services:
  - Guaranteed service: deterministic worst-case delay guarantees
    - RFC 2212
  - Controlled load service: similar to lightly-loaded best-effort
    - RFC 2211
- IntServ flow specification (*flowspec*):
  - Required service specification (*RSpec*)
    - Guaranteed / Controlled-load
    - Min-BW, Max-Delay, Max-Jitter, Max-Loss
    - Difficulty: for the network (admission control)
  - Traffic characteristics specification (*TSpec*)
    - Based on a token bucket envelope
    - Difficulty: for the user (traffic estimation)

Token/Leaky Bucket

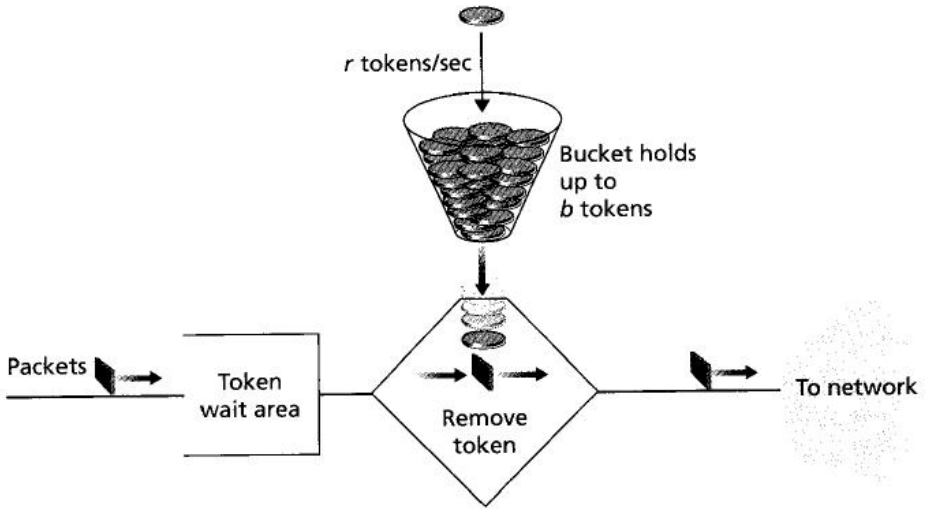
- Two parameters:
  - r: token arrival rate
  - b: bucket depth



- Modus operandi:
  - tokens arrive at the bucket at constant rate r
  - tokens are used by incoming packets
    - a packet of size M uses M tokens when sent
    - if no sufficient tokens exist: packet waits
  - bucket depth bounds number of tokens accumulated
    - when bucket is full: additional tokens are discarded

Token/Leaky Bucket

- Two parameters:
  - r: token arrival rate
  - b: bucket depth



- Properties of token/leaky bucket:
  - amount of data source transmits during interval I is bounded:

$$r\,|I| + b$$

  - long term average rate of traffic: at most r
  - maximum burst size generated by source: b

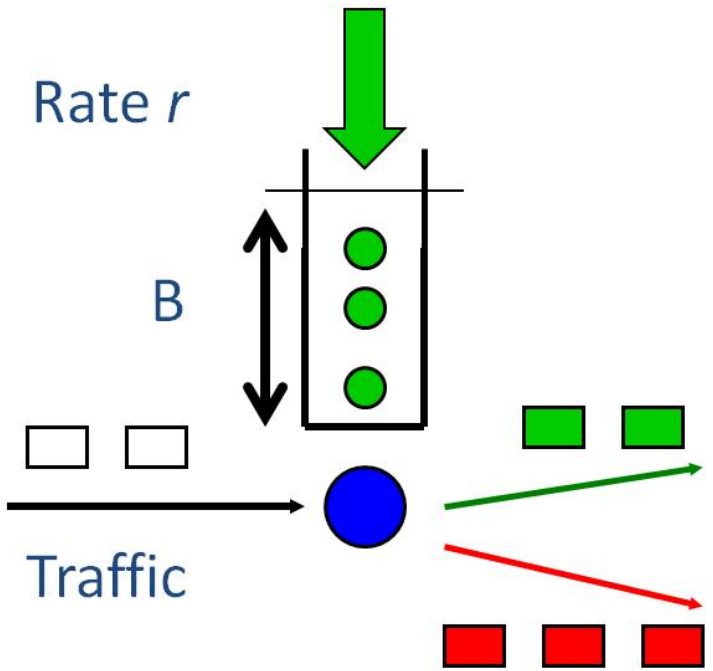
Flow Specification (*flowspec*)

- *TSpec* parameters:
  - Bucket rate (r) (B/s)
  - Peak rate (p) (B/s)
    - as large as source's line rate
  - Bucket depth (b) (B)
  - Minimum policed unit (m) (B)
    - any smaller packet is considered as m
  - Maximum packet size (M) (B)
- *RSpec* parameters:
  - Service rate (R) (B/s)
    - the main factor influencing E2E delay
  - Slack term (S) (ms)
    - bounding delay variability between hops

Guaranteed Service Only!

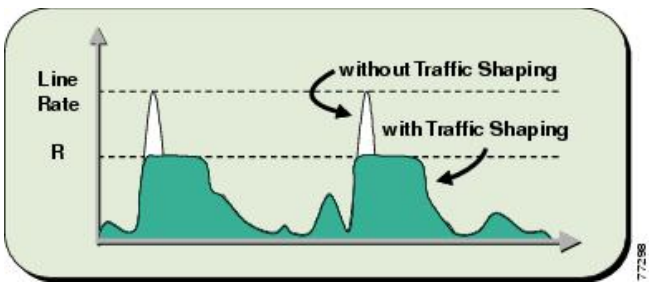
Policing, Marking and Shaping

- Policing and marking
  - Traffic conformance to *TSpec* is monitored at network edge
  - Nonconforming packets:
    - treated as best-effort
    - marked with drop priority
  - Should be done by the application...
    - might introduce additional delay otherwise (see below)



Policing, Marking and Shaping

- Policing and marking
  - Traffic conformance to *TSpec* is monitored at network edge
  - Nonconforming packets:
    - treated as best-effort
    - marked with drop priority
  - Should be done by the application...
    - might introduce additional delay otherwise (see below)
- Shaping
  - buffering performed along the path
  - occurs at multicast branch/merge points
  - necessary due to upstream vs. downstream links *TSpec* variability



Guaranteed Service

Delay Bounds

- Fluid model – single dedicated link:
  - assume $p \to \infty$ and $R \geq r$
  - E2E queuing delay:
$$\frac{b}{R}$$
  - essentially experienced by the last packet of a burst
- Fluid model – single dedicated link (arbitrary p):
  - E2E queuing delay:
$$\frac{b(p-R)}{R(p-r)} \quad (p > R \geq r)$$
  - in a burst: some packets served before last packet arrives
  - $p < R$: no delay
  - $r > R$: unbounded delay

Guaranteed Service

Delay Bounds

- Rate-dependent error $(C_{\text{sum}}, C_{\text{tot}})$
  - due to store-forward architecture
  - wait for packet last bit before forwarding
  - depends on packet size and transmission rate
  - sum over all nodes in the path
- Rate-independent error (system-dependent) $(D_{\text{sum}}, D_{\text{tot}})$
  - e.g., pipelining delay in router
  - route lookup, flow identification, etc.
- E2E queuing delay (with error terms):

$$\frac{(b-M)(p-R)}{R(p-r)} + \frac{M + C_{\text{tot}}}{R} + D_{\text{tot}} \quad p > R \geq r$$

- Additional delay factors:
  - propagation delay
  - shaping delay
  - end-systems processing delays

Controlled Load Service

- No strict reservations
  - no strict BW/delay assurance
- Suitable for adaptive applications (e.g., video)
- Considered as *better-than-best-effort*
- Similar to a lightly loaded network:
  - very low loss probability
  - very high probability of minimal delay
- What shouldn't happen:
  - long-term large delay
  - long-term loss
- Operation based on metering:
  - ensure guarantees to conforming flows
  - non-conforming traffic does not impact best-effort traffic
  - try and deliver non-conforming flow (subject to above)

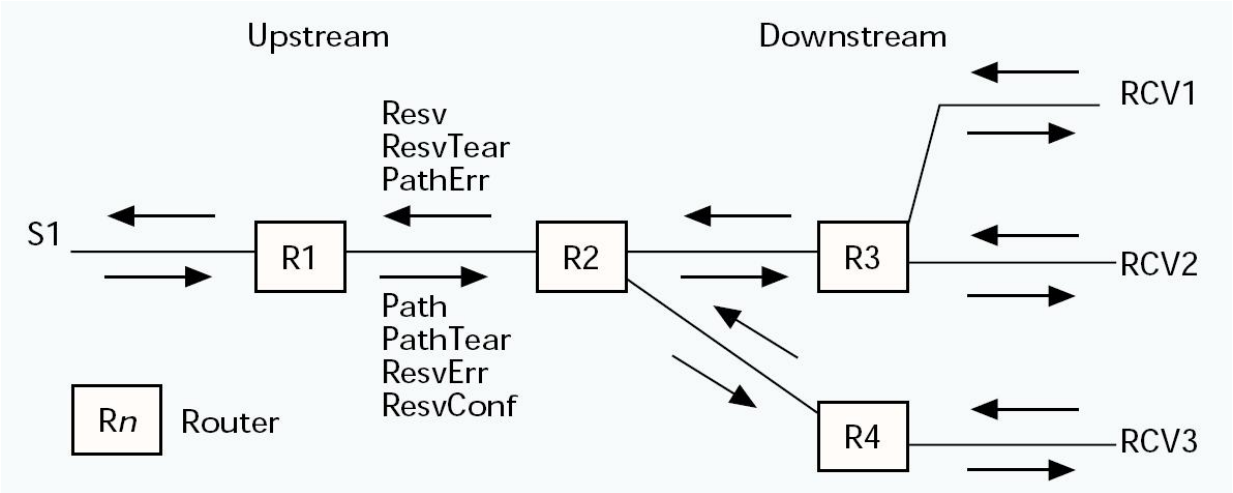
RSVP - Basic Features

- Resource reSerVation Protocol
  - RFC 2205
- Signaling protocol
  - allows users to communicate requirements to network
- Simplex reservation
  - establishes a reservation in only one direction
  - need to establish 2 reservations in two-way communication
- Receiver oriented
  - receivers decide what reservation is required
  - receivers initiate reservation

RSVP - Basic Features

- Routing independent
  - works with any unicast/multicast routing protocol
  - based on deployed routing mechanisms (BGP, OSPF, MPLS, etc.)
- Policy independent
  - e.g., independent of admission control policy
- Soft state
  - reservation state times out if not refreshed
  - enables robustness:
    - changing multicast group membership
    - changing network topologies
    - end-hosts crash

RSVP Messages



- Two main types of messages:
  - *PATH*
    - from sender to receiver
    - establishes the path
    - contains *TSpec*
  - *RESV*
    - from receiver back to sender
    - contains the actual reservation request
    - contains *TSpec* and *RSpec* (in GS)
- More messages
  - *PATHErr*
  - *RESVErr*
  - *PATHTear*
  - *RESVTear*
  - *RESVConf*

PATH Messages

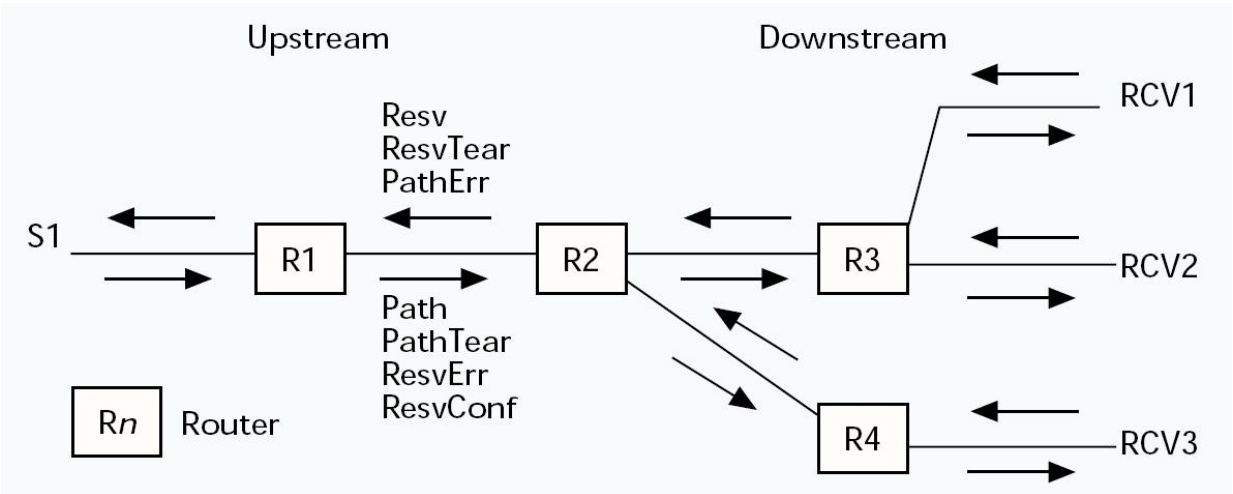
- Contain:
  - *Phop*
    - address of previous RSVP-capable node along path
    - used for sending the *RESV* message on the way back
  - *Sender Template*
    - IP address and port of sender
    - necessary for sending error
  - *TSpec*
  - *Adspec* (optional)
    - updated by nodes along path
    - One Pass with Advertising (OPWA) reservation model
    - used to collect info about path capabilities
    - e.g., path latency, min-BW, IntServ break bit, path MTU, etc.
    - for Guaranteed Service, also C_{tot}, C_{sum}, D_{tot}, D_{sum}.
    - enables receiver to pick *RSpec* parameters
- What do routers along path do?
  - init state with *Phop*, *Sender Template*, *TSpec*, and timer
  - update *Adspec* and *Phop*
  - forward *PATH* message over downstream links

RESV Messages

- Receiver uses *Adspec* and *TSpec* in *PATH* message
  - calculates required BW to ensure target delay
  - uses path MTU (not just *M* in sender *TSpec*)
- updates *flowspec*:
  - *TSpec*
    - may adjust parameters due to *Adspec*
  - *RSpec*
- *RESV* message also contains:
  - reservation style
  - *filterspec*
    - identifies sender, identical to *Sender Template*
  - *RSpec*
- What do routers along path do?
  - check *flowspec*: admission control + policy
  - init reservation state: *filterspec*, BW, etc.
  - possibly merge reservation (depends on reservation style)
  - forward *RESV* message over upstream links

Reservation Styles (Merging)

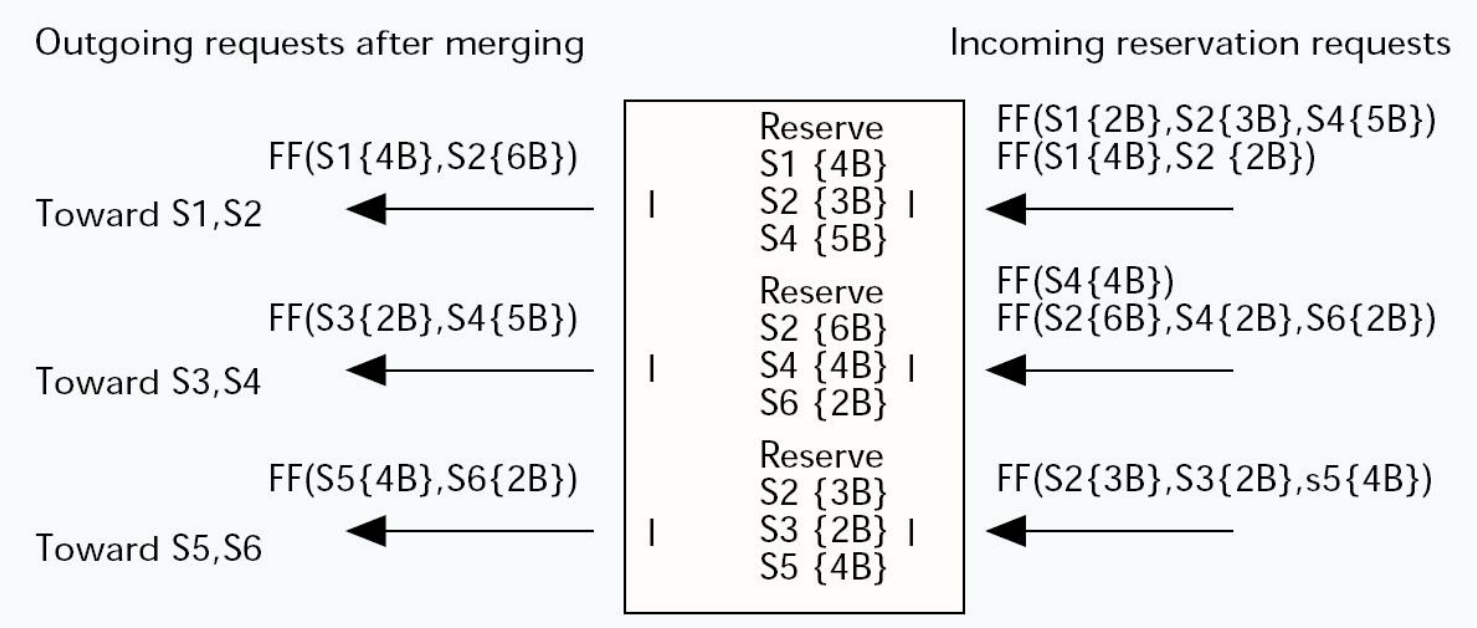
- Targeted towards multicast
  - recall video conferencing motivation...
  - single reservation style per session
- Determine how multiple reservation requests are merged
- Merging depends on:
  - sender (*filterspec*)
  - *flowspec*
  - router interface (in/out)
  - reservation style



Reservation Styles (Merging)

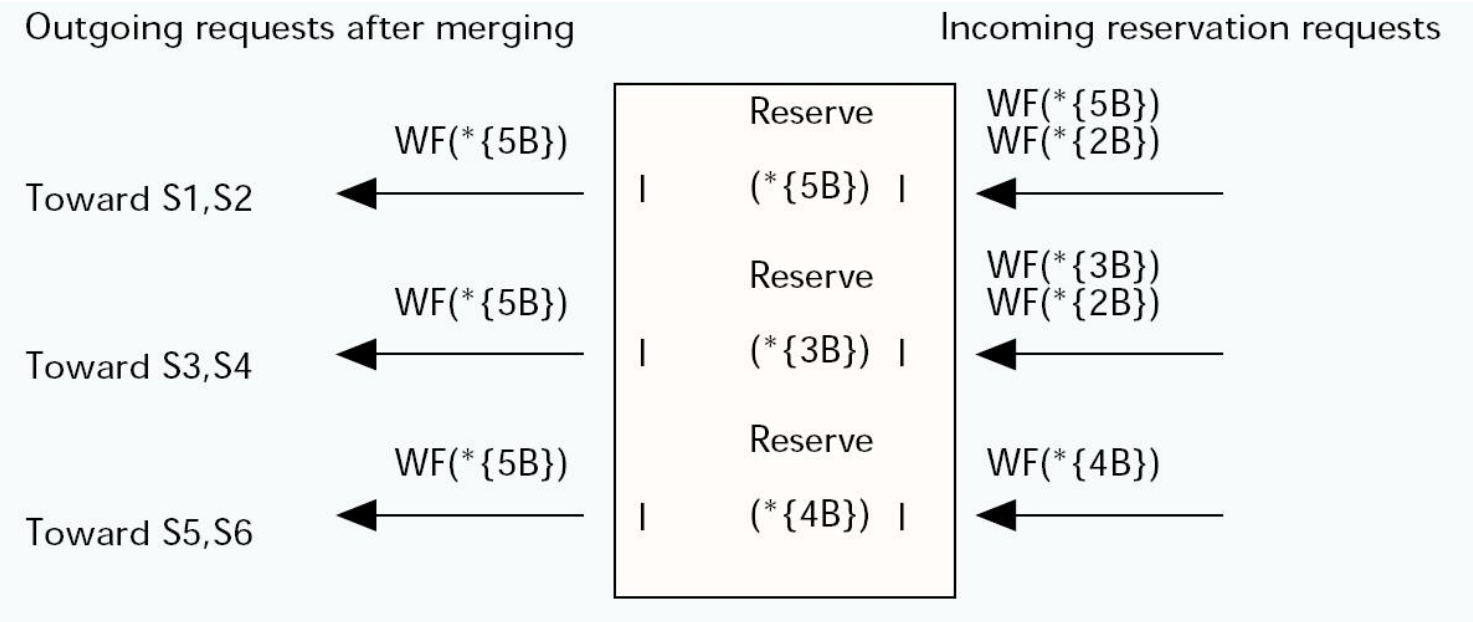
- Targeted towards multicast
  - recall video conferencing motivation...
  - single reservation style per session
- Determine how multiple reservation requests are merged
- Merging depends on:
  - sender (*filterspec*)
  - *flowspec*
  - router interface (in/out)
  - reservation style
- RSVP supports 3 reservation styles:
  - Fixed Filter (FF)
    - distinct reservation and explicit sender selection
  - Wildcard Filter (WF)
    - shared reservation and wildcard sender selection
  - Shared Explicit (SE)
    - shared reservation and explicit sender selection
- Format: *style* (*filterspec*{*flowspec*})

Fixed Filter (FF)



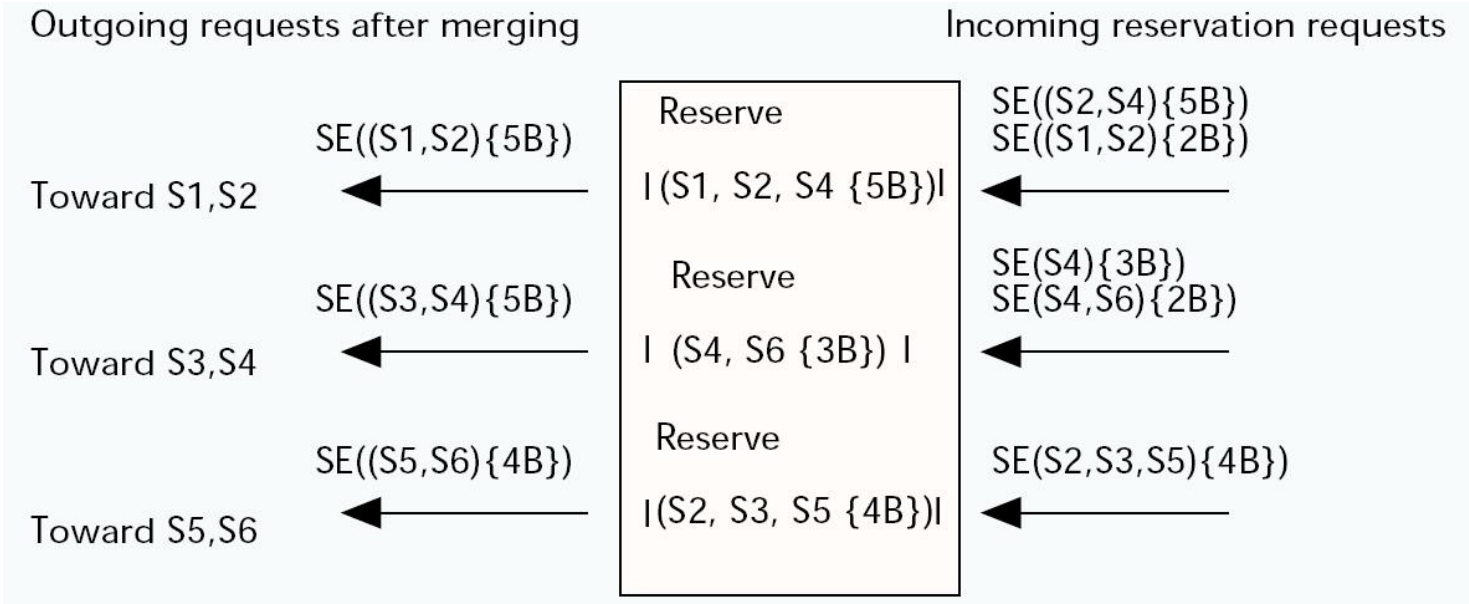
- Reservation made per-sender, per-downstream link
  - S2: 3 reservations (one/downstream link)
  - S4: 2 reservations
- Per-downstream link, per sender, reserve maximum over link
  - top, S2: $\max\{2B, 3B\} = 3B$
- Per-upstream link, per sender, request maximum reserved
  - top, S2: $\max\{3B, 6B, 3B\} = 6B$

Wildcard Filter (WF)



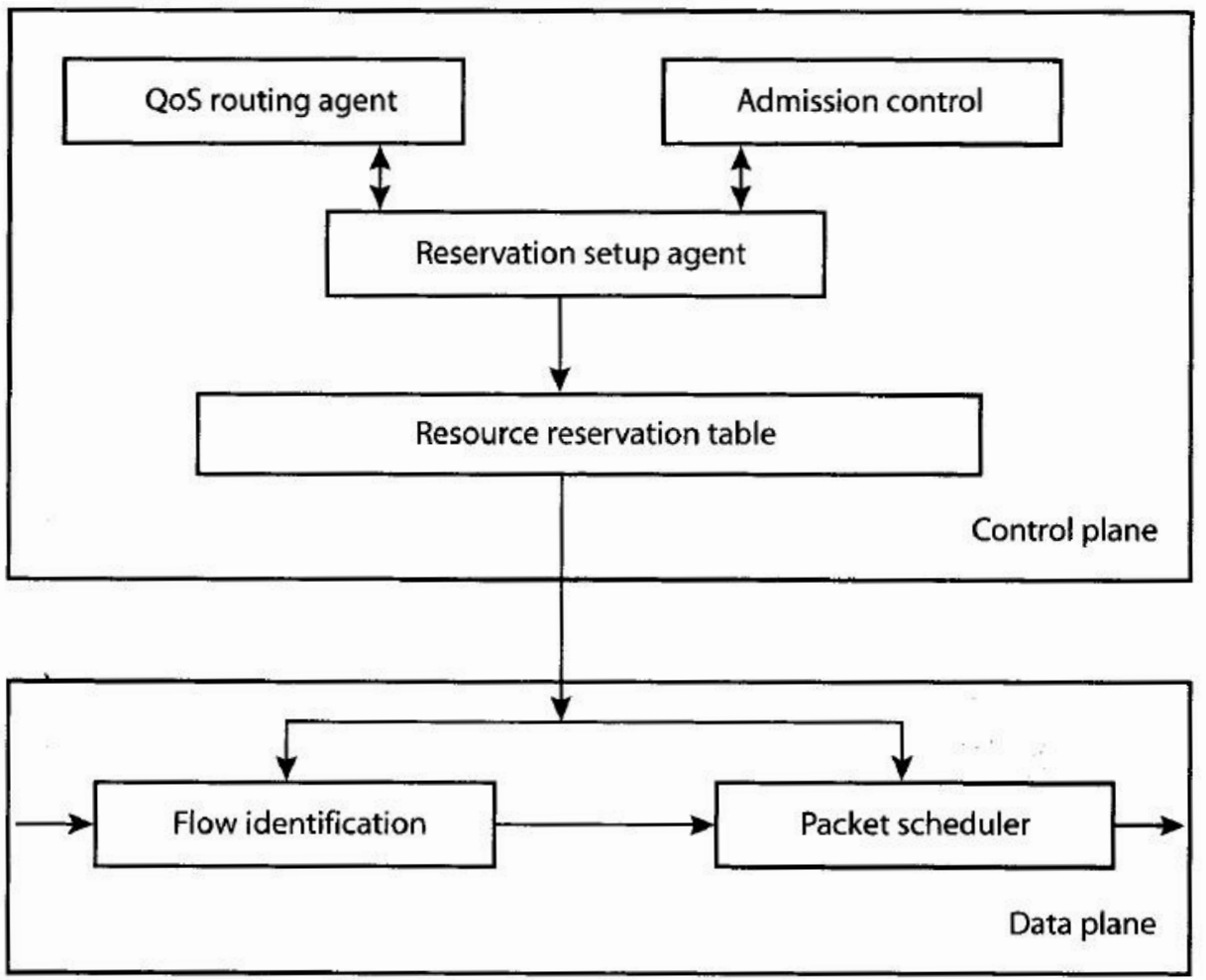
- Sender-oblivious
  - all senders can use reservations
- Reservation made per-downstream link, maximum requested
  - top: $\max\{5B, 2B\} = 5B$
  - middle: $\max\{2B, 3B\} = 3B$
- Per-upstream link, request maximum reserved
  - all: $\max\{5B, 3B, 4B\} = 5B$

Shared Explicit (SE)



- Explicit senders reservations
- Reservations made per-downstream link, maximum requested
  - all senders using that link can use reservations (union)
  - top: senders= {S1,S2,S4}, reservation= $\max\{5B, 2B\} = 5B$
- per-upstream link, request maximum reserved
  - all senders using that link can use reservations (union)
  - middle:
    - S3 has only $4B$ reserved on bottom downstream link
    - S3 shares requested $\max\{5B, 3B, 4B\} = 5B$

Summary – Router Architecture



References

- Kurose and Ross, "*Computer Networking: A Top-Down Approach*", 5th ed., Addison-Wesley, 2010.
- Peterson and Davie, "*Computer Networks: A Systems Approach*", 4th ed., Morgan Kaufmann, 2007.
- Wang, "*Internet QoS: Architectures and Mechanisms for Quality of Service*", Morgan Kaufmann, 2001.
- Farrel, "*The Internet and Its Protocols: A Comparative Approach*". Morgan Kaufmann, 2004.
- White, RSVP and Integrated Services in the Internet: A Tutorial. IEEE Communications Magazine, May 1997