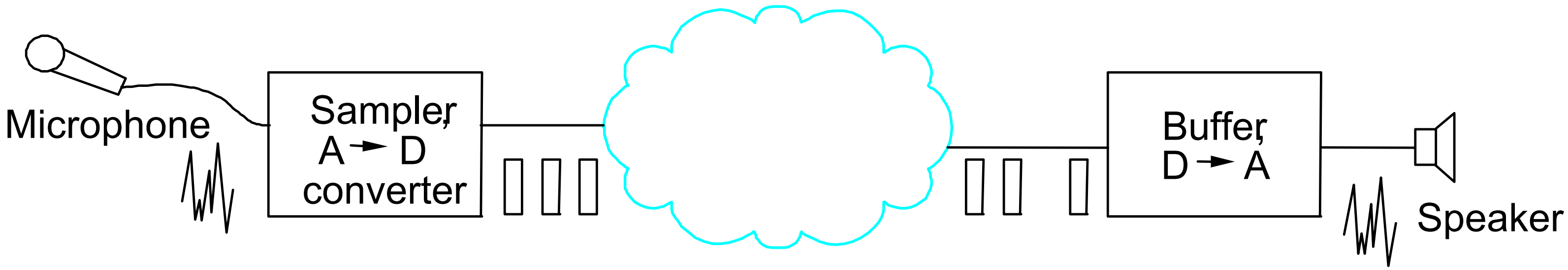
Quality of Service

Outline

- Realtime Applications
- Integrated Services
- Differentiated Services

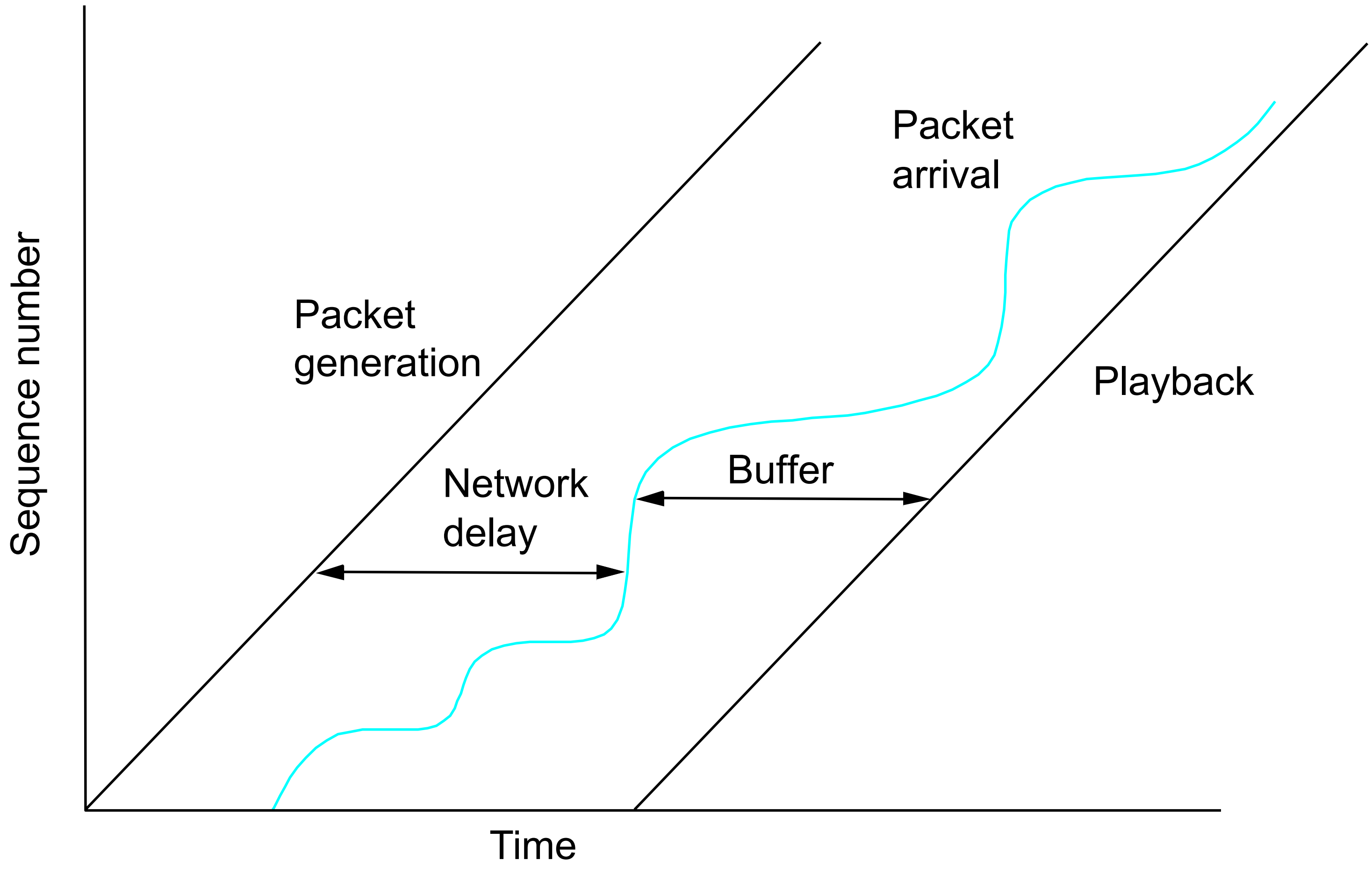
Realtime Applications

- Require "deliver on time" assurances
  - must come from *inside* the network

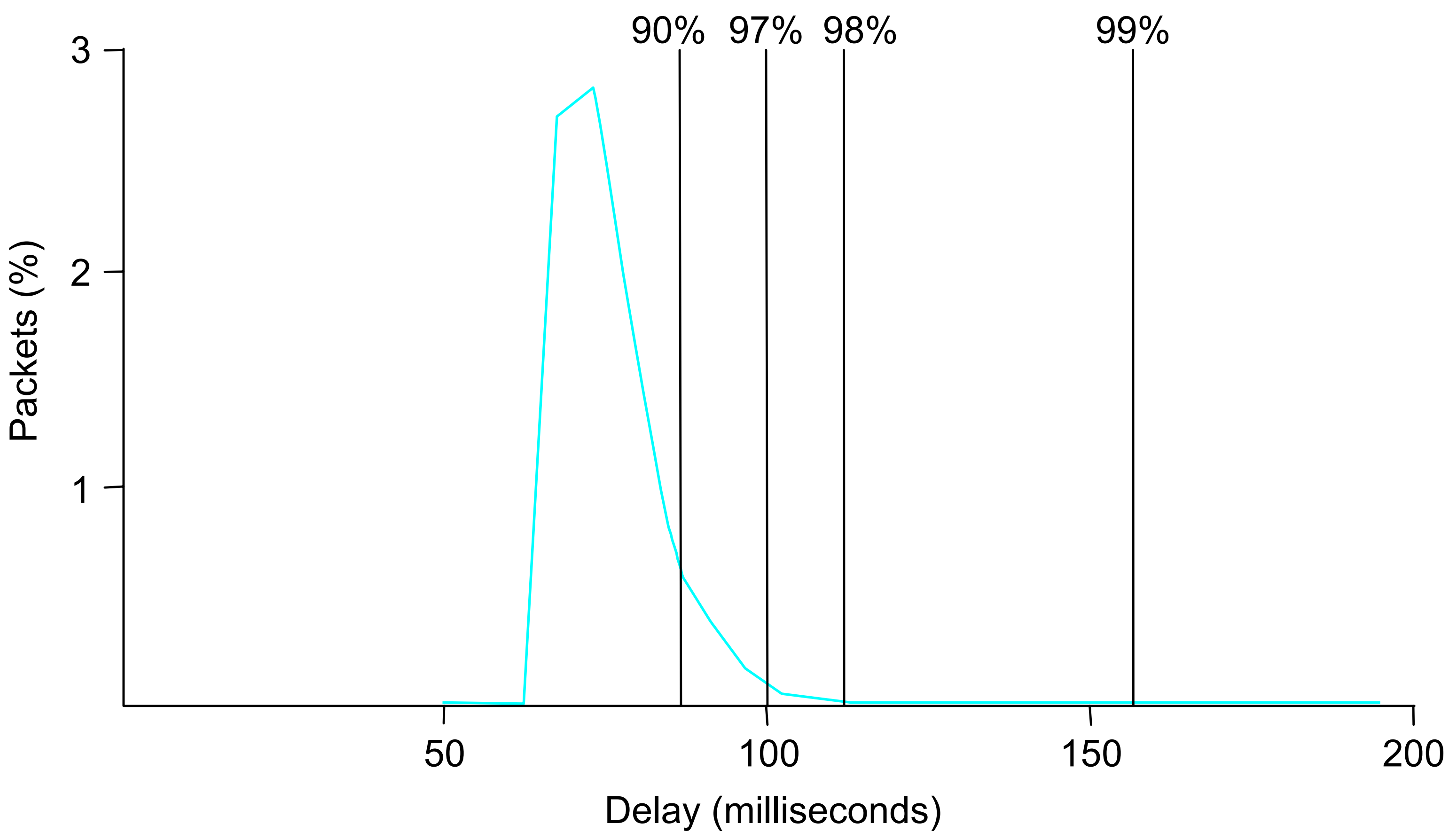


- Example application (audio)
  - sample voice once every 125us
  - each sample has a *playback time*
  - packets experience variable delay in network
  - add constant factor to playback time: *playback point*

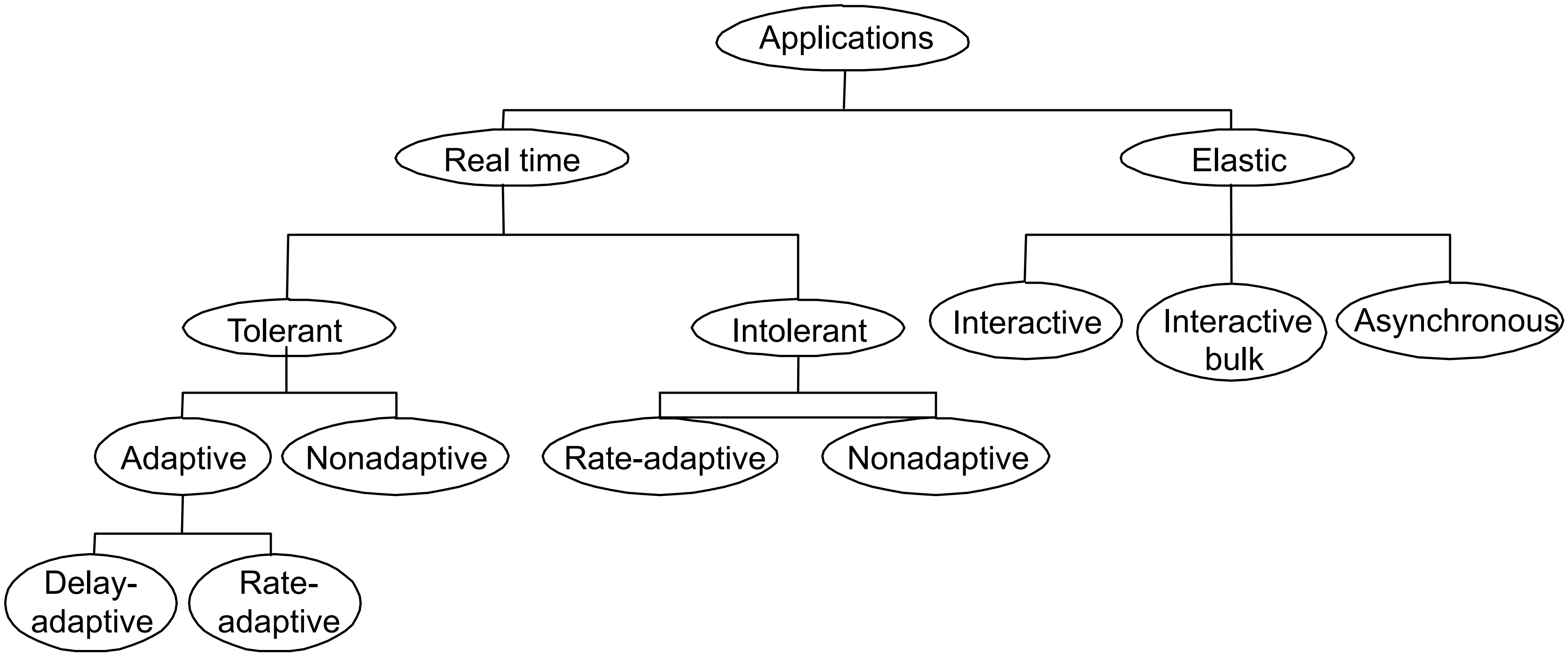
Playback Buffer



Example Distribution of Delays



Taxonomy



Applications
Real time
Elastic
Tolerant
Intolerant
Interactive
Interactive bulk
Asynchronous
Adaptive
Nonadaptive
Rate-adaptive
Nonadaptive
Delay-adaptive
Rate-adaptive

Integrated Services

- Service Classes
  - guaranteed
  - controlled-load
- Mechanisms
  - signalling protocol
  - admission control
  - policing
  - packet scheduling

Flowspec

- ***Rspec:*** describes service requested from network
  - controlled-load: none
  - guaranteed: delay target
- ***Tspec:*** describes flow's traffic characteristics
  - average bandwidth + burstiness: *token bucket* filter
  - token rate *r*
  - bucket depth *B*
  - must have a token to send a byte
  - must have *n* tokens to send *n* bytes
  - start with no tokens
  - accumulate tokens at rate of *r* per second
  - can accumulate no more than *B* tokens

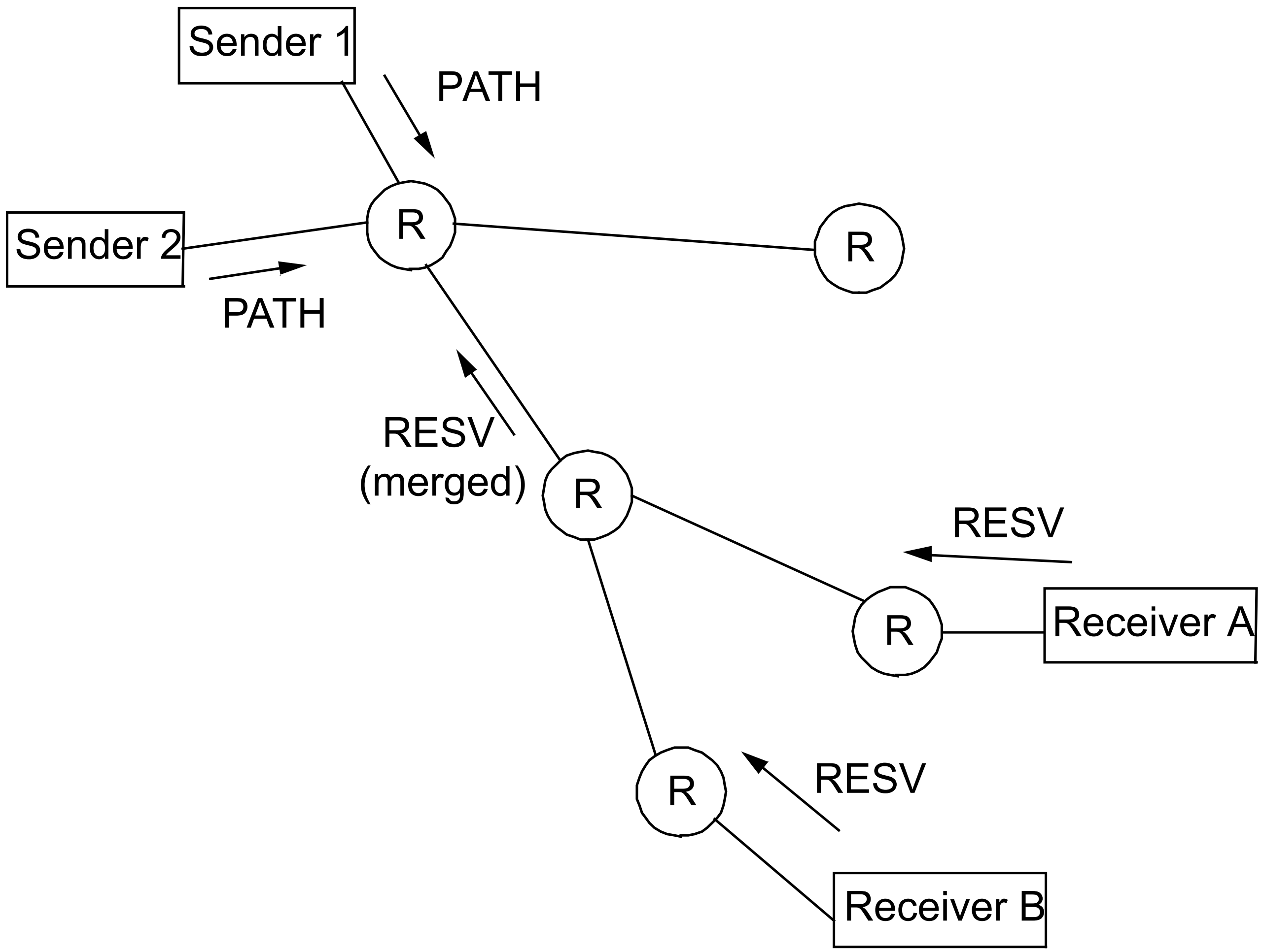
Per-Router Mechanisms

- Admission Control
  - decide if a new flow can be supported
  - answer depends on service class
  - not the same as *policing* or *shaping*
- Packet Processing
  - classification: associate each packet with the appropriate reservation
  - scheduling: manage queues so each packet receives the requested service

Reservation Protocol

- Called signaling in ATM
- Proposed Internet standard: RSVP
- Consistent with robustness of today's connectionless model
- Uses soft state (refresh periodically)
- Designed to support multicast
- Receiver-oriented
- Two messages: PATH and RESV
- Source transmits PATH messages every 30 seconds
- Destination responds with RESV message
- Merge requirements in case of multicast
- Can specify number of speakers

RSVP Example



RSVP versus ATM (Q.2931)

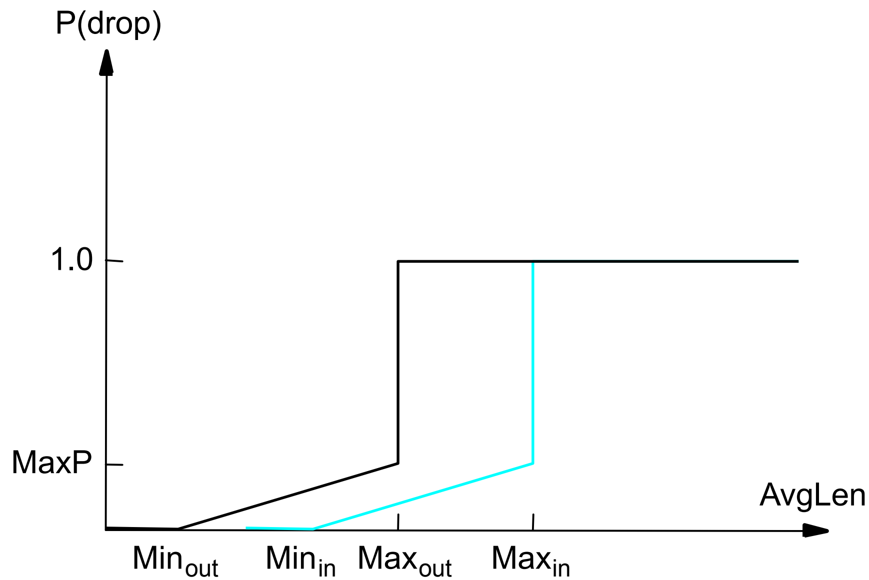
- RSVP
  - receiver generates reservation
  - soft state (refresh/timeout)
  - separate from route establishment
  - QoS can change dynamically
  - receiver heterogeneity
- ATM
  - sender generates connection request
  - hard state (explicit delete)
  - concurrent with route establishment
  - QoS is static for life of connection
  - uniform QoS to all receivers

Differentiated Services

- Problem with Integrated Services
  - Scalability – amount of state in core routers
  - Classification cost in core routers
  - Limited service models (can't say class A gets better service than class B, e.g.)
- Differentiated Services (DiffServ)
  - RFC 2474/2475
  - Goal is scalability
  - Flexible differentiation based on a set of components
- Key to the approach
  - *packet* classification and labelling at network *edge*
  - Forwarding "priority" according to label in the network core – per hop behavior
    - Core routers don't have per-flow state

Example mechanism: RIO

- RED with In and Out
- Mechanism
  - packets: ‘in’ and ‘out’ bit
  - edge routers: tag packets
  - core routers: RIO
    (RED with In and Out)



Per-Hop Behaviors

- a description of the externally observable forwarding behavior of a DiffServ node applied to a particular DiffServ behavior aggregate (class)
  - Doesn't define the implementation
- Example descriptions (generic)
  - Class A gets at least x% of link bandwidth
  - Class A receives priority over Class B

Expedited Forwarding PHB

- RFC 2598
- An EF class receives a guaranteed minimum bandwidth
  - Requires isolation from other traffic
  - Abstractly: we've created a virtual link with the required minimum bandwidth for the EF class
- Example mechanism for achieving: WFQ

Assured Forwarding PHB

- RFC 2597
- Traffic is divided into 4 classes each with 3 “drop-preference” categories.
- Each class receives a guaranteed minimum amount of bandwidth and buffering
- If you were an ISP how could make a product out of this kind of mechanism?

Core Stateless Fair Queuing

- DiffServ-like marking at the network edge
  – Packets in a flow are marked with their observed arrival rate (possibly weighted)
- Core routers
  – Preferentially drop packets with high arrival rates
  – Relabel packets according to their actual departure rate
- Relies on adaptation to packet loss by the sender to achieve fair allocations
  – But those who don't play by the rules can be punished
- Stoica, Shenker and Zhang. *Core-Stateless Fair Queueing: Achieving Approximately Fair Bandwidth Allocations in High Speed Networks*, in ACM SIGCOMM, August 1998.

Not playing by the rules – unfriendly flows

- Fire hose application
  - Gets x% of the available value from x% of the packets it sends
  - No motivation to reduce sending rate to match fair share
  - Network should punish this behavior by reducing packet delivery to below fair share
- Greedy AIMD
  - A bigger problem with FIFO/Tail Drop than with anything that approximates FQ

Quick Essay

- Consider the grading guide for project 2.
- Overall, was having the guide helpful, compared with your experience in project 1 and other projects you have done?
- What in the grading guide was most helpful to you in preparing project 2?
- What in the grading guide was least helpful to you in preparing project 2?
- What would you most like to see changed in the grading guide for project 3?