CS 640 Introduction to Computer Networks

Lecture27

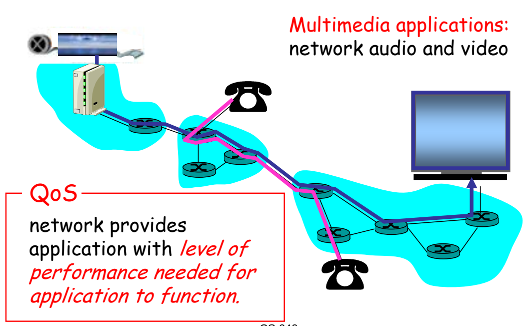
CS 640

Today's lecture

- Quality of Service
  - Requirements of multimedia applications
  - Scheduling and policing mechanisms
  - Architectures for enabling QoS
    - IntServ
    - DiffServ

CS 640

Quality of Service: What is it?



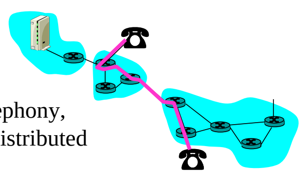
CS 640

Performance Requirements

Requirement: deliver data in "timely" manner

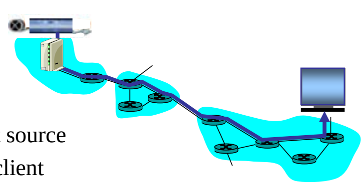
- interactive multimedia: short end-end delay
  - e.g., IP telephony, teleconferencing, virtual worlds
  - excessive delay impairs human interaction
- streaming (non-interactive) multimedia:
  - data must arrive in time for "smooth" playout
  - late arriving data introduces gaps in rendered audio/video
- reliability: 100% reliability not always required

Interactive, Real-Time Multimedia



- applications: IP telephony, video conference, distributed interactive worlds
- end-end delay requirements:
  - video: < 150 msec acceptable
  - audio: < 150 msec good, < 400 msec OK
  - add application-level (packetization) and network delay
  - higher delays noticeable, impair interactivity

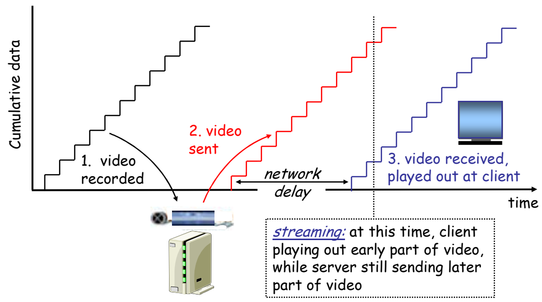
Streaming Multimedia



Streaming:

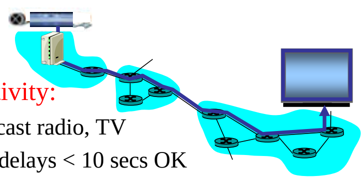
- media stored at source
- transmitted to client
- streaming: client playout begins *before* all data has arrived
- timing constraint for still-to-be transmitted data: in time for playout

Streaming: what is it?



CS 640

Streaming Multimedia (more)



Types of interactivity:

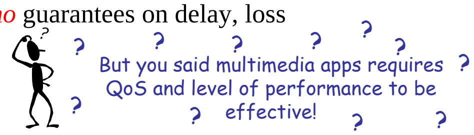
- *none:* like broadcast radio, TV
  - initial startup delays < 10 secs OK
- *VCR-functionality:* client can pause, rewind, FF
  - 1-2 sec until command effect OK
  - timing constraint for still-to-be transmitted data: in time for playout

CS 640

Multimedia Over Today's Internet

TCP/UDP/IP: "best-effort service"

- *no* guarantees on delay, loss



Today's Internet multimedia applications use application-level techniques to mitigate (as best possible) effects of delay, loss

CS 640

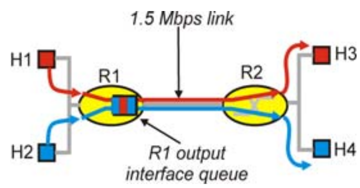
Improving QoS in IP Networks

Thus far: "making the best of best effort"

Future: next generation Internet with QoS guarantees

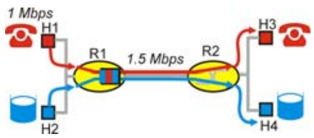
- RSVP: signaling for resource reservations
- Differentiated Services: differential guarantees
- Integrated Services: firm guarantees

- simple model for sharing and congestion studies:



Principles for QOS Guarantees

- Example: 1Mbps IP phone, ftp share 1.5 Mbps link.
  - bursts of FTP can congest router, cause audio loss
  - want to give priority to audio over FTP

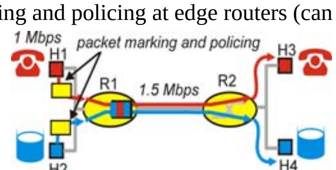


Principle 1

packet marking needed for router to distinguish between different classes; and new router policy to treat packets accordingly

Principles for QOS Guarantees

- what if applications misbehave (audio sends higher than declared rate)
  - policing: force source adherence to bandwidth allocations
- Marking and policing at edge routers (can keep state)

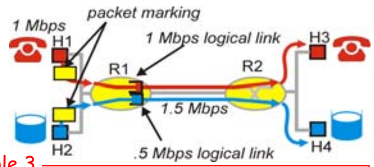


Principle 2

provide protection (*isolation*) for one class from others

Principles for QOS Guarantees

- Allocating *fixed* (non-sharable) bandwidth: *inefficient* use of network if flow doesn't use its allocation



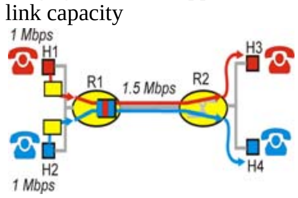
Principle 3

While providing isolation, it is desirable to use resources as efficiently as possible

CS 640

Principles for QOS Guarantees

- *Basic fact of life:* can not support traffic demands beyond link capacity

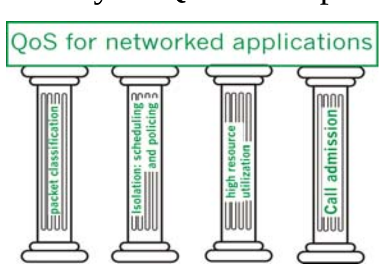


Principle 4

Call Admission: flow declares its needs, network may block call (e.g., busy signal) if it cannot meet needs

CS 640

Summary of QoS Principles

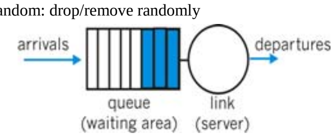


Let's next look at mechanisms for achieving this ....

CS 640

Scheduling and Policing Mechanisms

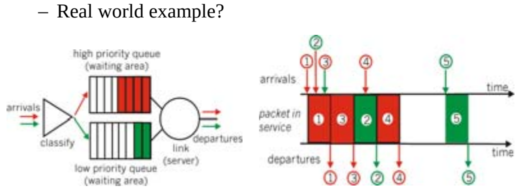
- scheduling: choose next packet to send on link
- FIFO (first in first out) scheduling: send in order of arrival to queue
  - real-world example?
  - discard policy: packet arrives queue full – what to discard?
    - tail drop: drop arriving packet
    - priority: drop/remove on priority basis
    - random: drop/remove randomly



Scheduling Policies: more

Priority scheduling: transmit highest priority packet

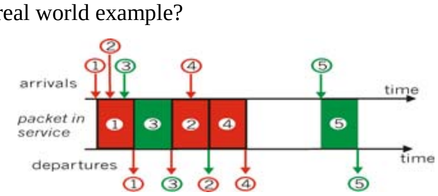
- multiple *classes*, with different priorities
  - class may depend on marking or other header info, e.g. IP source/dest, port numbers, etc..
  - Real world example?



Scheduling Policies: still more

Round robin scheduling:

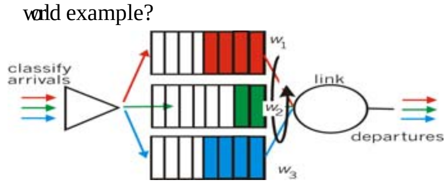
- multiple classes
- cyclically scan class queues, serving one from each class (if available)
- real world example?



Scheduling Policies: still more

Weighted Fair Queuing:

- generalized Round Robin
- each class gets weighted amount of service in each cycle
- real-world example?



Policing Mechanisms

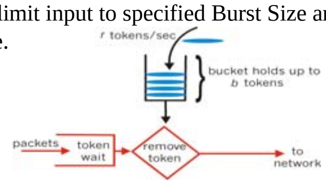
Goal: limit traffic to not exceed declared parameters

Three common-used criteria:

- *(Long term) Average Rate:* how many pkts can be sent per unit time (in the long run)
  - crucial question – what is the interval length: 100 packets per sec or 6000 packets per min have same average!
- *Peak Rate:* e.g., 6000 pkts per min. (ppm) avg.; 9000 ppm peak rate
- *(Max.) Burst Size:* max. number of pkts sent consecutively (with no intervening idle)

Policing Mechanisms

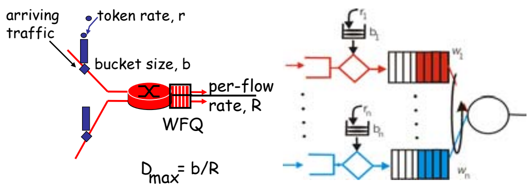
Token Bucket: limit input to specified Burst Size and Average Rate.



- bucket can hold b tokens
- tokens generated at rate *r token/sec* unless bucket full
- *over interval of length t: number of packets admitted less than or equal to (r t + b).*

Policing Mechanisms

- token bucket, WFQ combine to provide guaranteed upper bound on delay, i.e., *QoS guarantee*!



CS 640

IETF Integrated Services

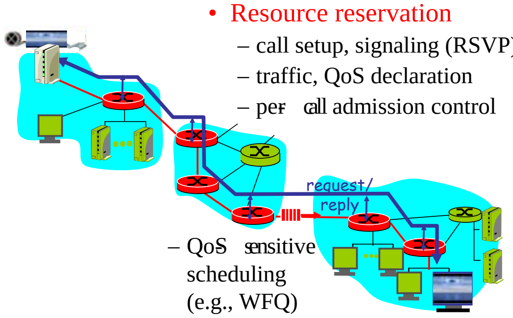
- architecture for providing QOS guarantees in IP networks for individual application sessions
- resource reservation: routers maintain state info (a la VC) of allocated resources, QoS req's
- admit/deny new call setup requests:

Question: can newly arriving flow be admitted with performance guarantees while not violated QoS guarantees made to already admitted flows?

CS 640

Intserv: QoS guarantee scenario

- Resource reservation
  - call setup, signaling (RSVP)
  - traffic, QoS declaration
  - per-call admission control



- QoS-sensitive scheduling (e.g., WFQ)

CS 640

Call Admission

Arriving session must :

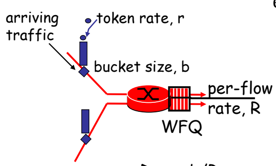
- declare its QoS requirement
  - Rspec: defines the QoS being requested
- characterize traffic it will send into network
  - Tspec: defines traffic characteristics
- signaling protocol: needed to carry Rspec and Tspec to routers (where reservation is required)
  - RSVP

CS 640

Intserv QoS: Service models [rfc2211, rfc 2212]

Guaranteed service:

- worst case traffic arrival: leaky-bucket-policed source
- simple (mathematically provable) *bound* on delay [Parekh 1992, Cruz 1988]



$D_{max} = b/R$

Controlled load service:

- "a quality of service closely approximating the QoS that same flow would receive from an unloaded network element."

CS 640

IETF Differentiated Services

Concerns with Intserv:

- Scalability: signaling, maintaining per-flow router state difficult with large number of flows
- Flexible Service Models: Intserv has only two classes. Also want “qualitative” service classes
  - “behaves like a wire”
  - relative service distinction: Platinum, Gold, Silver

Diffserv approach:

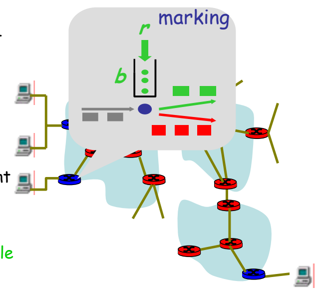
- Simple functions in network core, relatively complex functions at edge routers (or hosts)
- Don’t define define service classes, provide functional components to build service classes

CS 640

Diffserv Architecture

Edge router:
- per-flow traffic management
- marks packets as in-profile and out-profile

Core router:
- per class traffic management
- buffering and scheduling based on marking at edge
- preference given to in-profile packets
- Assured Forwarding

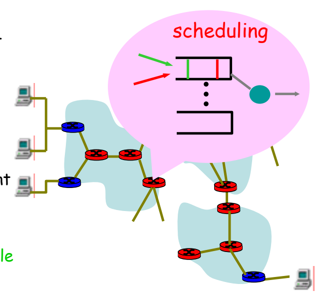


CS 640

Diffserv Architecture

Edge router:
- per-flow traffic management
- marks packets as in-profile and out-profile

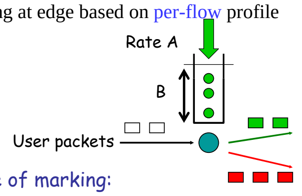
Core router:
- per class traffic management
- buffering and scheduling based on marking at edge
- preference given to in-profile packets
- Assured Forwarding



CS 640

Edge-router Packet Marking

- profile: pre-negotiated rate A, bucket size B
- packet marking at edge based on per-flow profile



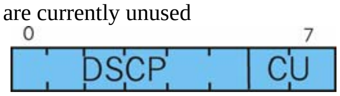
Possible usage of marking:

- class-based marking: packets of different classes marked differently
- intra-class marking: conforming portion of flow marked differently than non-conforming one

CS 640

Classification and Conditioning

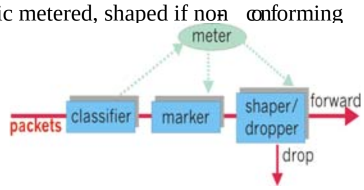
- Packet is marked in the Type of Service (TOS) in IPv4, and Traffic Class in IPv6
- 6 bits used for Differentiated Service Code Point (DSCP) and determine PHB that the packet will receive
- 2 bits are currently unused



Classification and Conditioning

may be desirable to limit traffic injection rate of some class:

- user declares traffic profile (eg, rate, burst size)
- traffic metered, shaped if non-conforming



Forwarding (PHB)

- PHB result in a different observable (measurable) forwarding performance behavior
- PHB does not specify what mechanisms to use to ensure required PHB performance behavior
- Examples:
  - Class A gets x% of outgoing link bandwidth over time intervals of a specified length
  - Class A packets leave first before packets from class B

Forwarding (PHB)

PHBs being developed:

- Expedited Forwarding: pkt departure rate of a class equals or exceeds specified rate
  - logical link with a minimum guaranteed rate
- Assured Forwarding: 4 classes of traffic
  - each guaranteed minimum amount of bandwidth
  - each with three drop preference partitions

Multimedia Networking: Summary

- multimedia applications and requirements
- making the best of today's best effort service
- scheduling and policing mechanisms
- next generation Internet: Intserv, RSVP, Diffserv