# Chapter 6
# Multimedia Networking

A note on the use of these ppt slides:
We're making these slides freely available to all (faculty, students, readers). They're in powerpoint form so you can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part. In return for use, we only ask the following:
- If you use these slides (e.g., in a class) in substantially unaltered form, that you mention their source (after all, we'd like people to use our book!)
- If you post any slides in substantially unaltered form on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

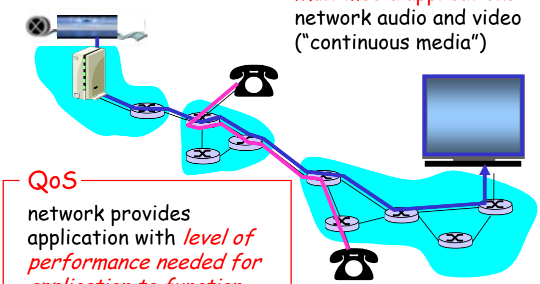
Thanks and enjoy! JFK / KWR



*Computer Networking: A Top Down Approach Featuring the Internet,*
2nd edition.
Jim Kurose, Keith Ross
Addison-Wesley, July 2002.

## Multimedia, Quality of Service: What is it?

Multimedia applications:
network audio and video ("continuous media")



QoS
network provides application with *level of performance needed for application to function.*

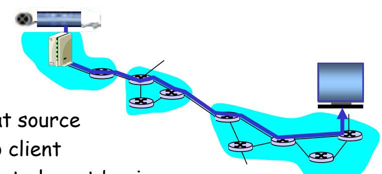
## MM Networking Applications

Classes of MM applications:
1) Streaming stored audio and video
2) Streaming live audio and video
3) Real-time interactive audio and video

Fundamental characteristics:
- Typically **delay sensitive**
  - end-to-end delay
  - delay jitter
- But **loss tolerant**: infrequent losses cause minor glitches
- Antithesis of data, which are loss intolerant but delay tolerant.

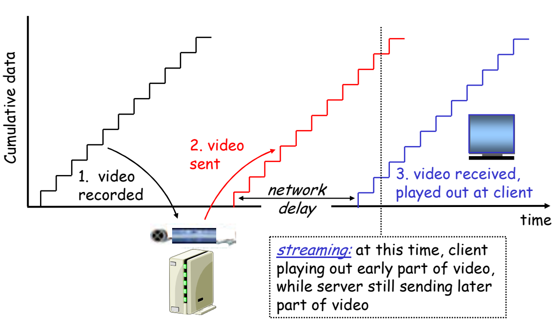
**Jitter** is the variability of packet delays within the same packet stream

## Streaming Stored Multimedia

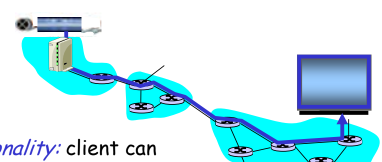


Streaming:
- media stored at source
- transmitted to client
- streaming: client playout begins *before* all data has arrived
  - timing constraint for still-to-be transmitted data: in time for playout

## Streaming Stored Multimedia: What is it?




## Streaming Stored Multimedia: Interactivity



- *VCR-like functionality:* client can pause, rewind, FF, push slider bar
  - 10 sec initial delay OK
  - 1-2 sec until command effect OK
  - RTSP often used (more later)
- timing constraint for still-to-be transmitted data: in time for playout

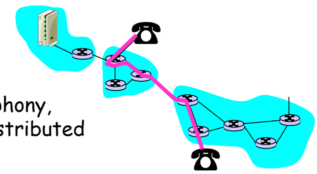
## Streaming Live Multimedia

Examples:
- Internet radio talk show
- Live sporting event

Streaming
- playback buffer
- playback can lag tens of seconds after transmission
- still have timing constraint

Interactivity
- fast forward impossible
- rewind, pause possible!

## Interactive, Real-Time Multimedia



applications: IP telephony, video conference, distributed interactive worlds

end-end delay requirements:
- audio: < 150 msec good, < 400 msec OK
  - includes application-level (packetization) and network delays
  - higher delays noticeable, impair interactivity

session initialization
- how does callee advertise its IP address, port number, encoding algorithms?

## Multimedia Over Today's Internet

TCP/UDP/IP: "best-effort service"
- *no* guarantees on delay, loss

? ? ? ? ? ? But you said multimedia apps requires ? QoS and level of performance to be ? effective! ? ? ?

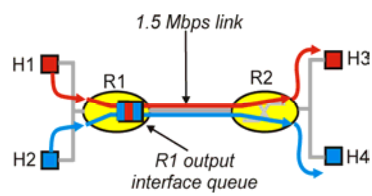
Today's Internet multimedia applications use application-level techniques to mitigate (as best possible) effects of delay, loss

## Improving QOS in IP Networks

Thus far: "making the best of best effort"

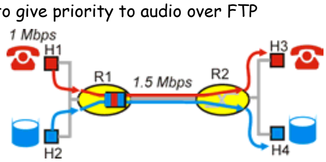
Future: next generation Internet with QoS guarantees
- RSVP: signaling for resource reservations
- Differentiated Services: differential guarantees
- Integrated Services: firm guarantees

simple model for sharing and congestion studies:




## Principles for QOS Guarantees

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



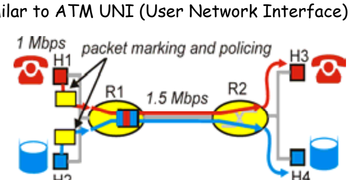

Principle 1

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

## Principles for QOS Guarantees (more)

what if applications misbehave (audio sends higher than declared rate)
- policing: force source adherence to bandwidth allocations

marking and policing at network edge:
- similar to ATM UNI (User Network Interface)

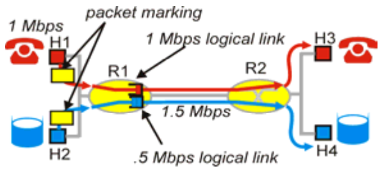



Principle 2

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

## Principles for QOS Guarantees (more)

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

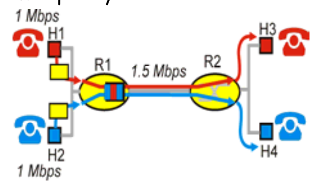



**Principle 3**

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

## Principles for QOS Guarantees (more)

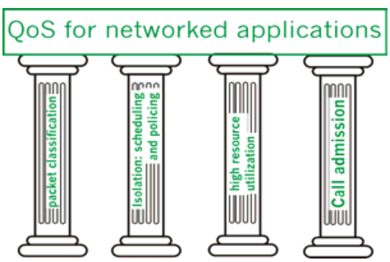
*Basic fact of life:* can not support traffic demands beyond link capacity




**Principle 4**

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

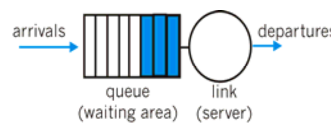
## Summary of QoS Principles




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

## Scheduling And Policing Mechanisms

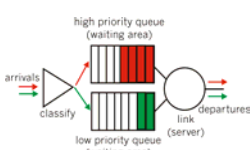
- scheduling: choose next packet to send on link
- FIFO (first in first out) scheduling: send in order of arrival to queue
  - real-world example?
  - discard policy: if packet arrives to full queue: who to discard?
    - Tail drop: drop arriving packet
    - priority: drop/remove on priority basis
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

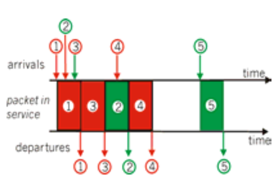



## Scheduling Policies: more

Priority scheduling: transmit highest priority queued 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?

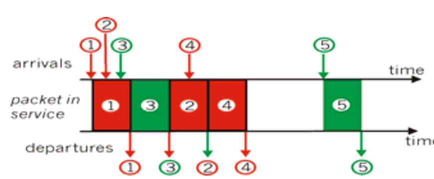



## Scheduling Policies: still more

round robin scheduling:

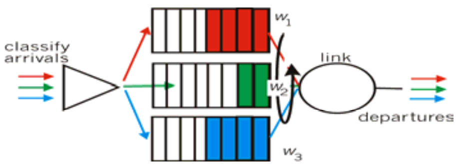
- multiple classes
- cyclically scan class queues, serving one from each class (if available)
- real world example?



## 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

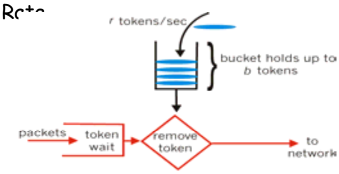
Goal: limit traffic to not exceed declared parameters

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.; 1500 ppm peak rate
- *(Max.) Burst Size:* max. number of pkts sent consecutively (with no intervening idle)

## Policing Mechanisms

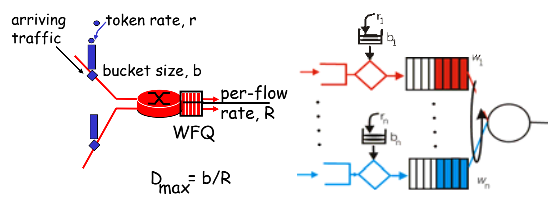
Token Bucket: limit input to specified Burst Size and Average Rate.




- 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 (more)

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



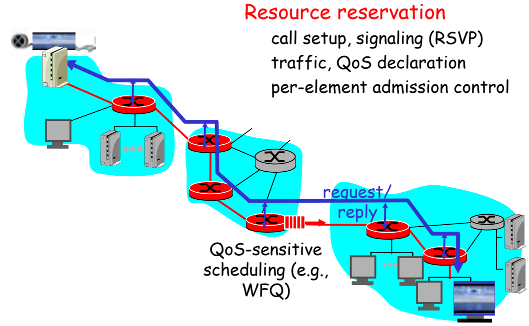

## IETF Integrated Services

- 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?

## Intserv: QoS guarantee scenario

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



## Call Admission

Arriving session must :

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

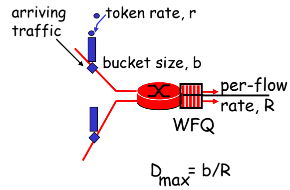
## Intserv QoS: Service models [rfc2211, rfc 2212]

**Guaranteed service:**

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

**Controlled load service:**

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




## 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)
- Do't define define service classes, provide functional components to build service classes

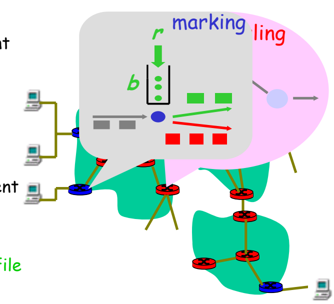
## Diffserv Architecture

Edge router:

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

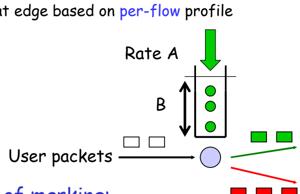
Core router:

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




## Edge-router Packet Marking

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

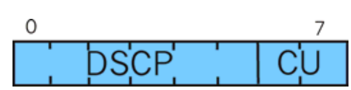



Possible usage of marking:

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

## 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