Master Course Computer Networks IN2097

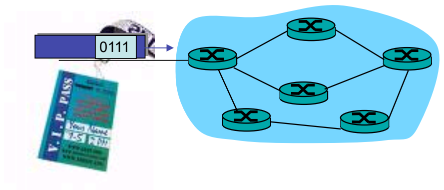
Prof. Dr.-Ing. Georg Carle
Christian Grothoff, Ph.D.

Chair for Network Architectures and Services
Institut für Informatik
Technische Universität München
http://www.net.in.tum.de

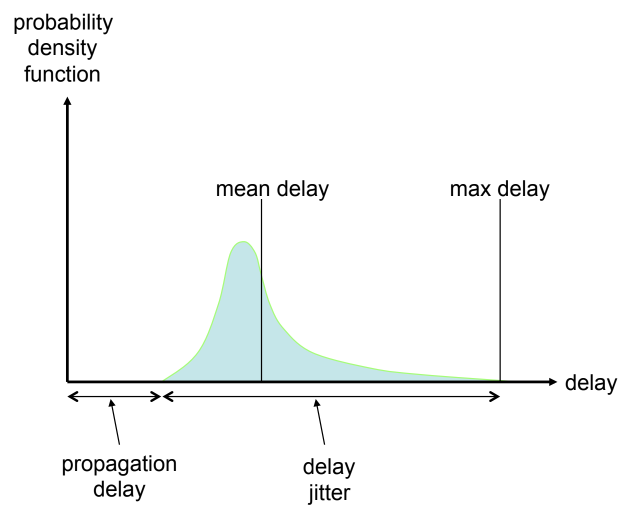
Quality of Service Support

Providing Multiple Classes of Service

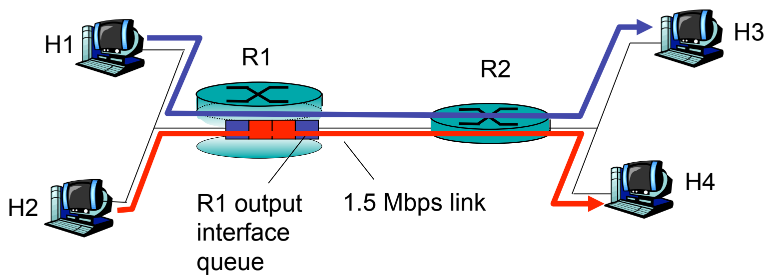
- Traditional Internet approach: making the best of best effort service
  - one-size fits all service model
- Alternative approach: multiple classes of service
  - partition traffic into classes
  - network treats different classes of traffic differently (analogy: VIP service vs regular service)
- granularity:
  differential service among multiple classes, not among individual connections
- history:
  ToS bits in IP header



Delay Distributions

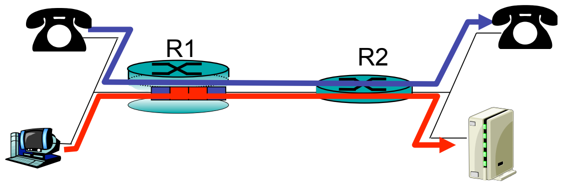


Multiple classes of service: scenario



Scenario 1: mixed FTP and audio

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

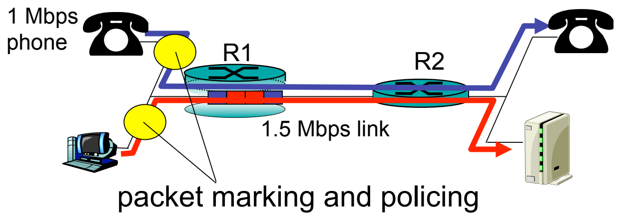


Principle 1

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)

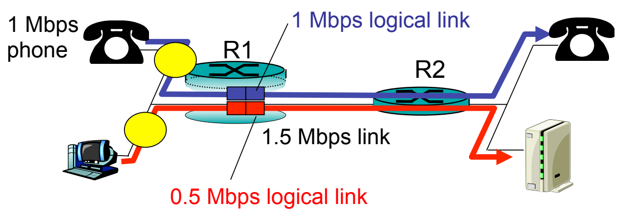


Principle 2

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

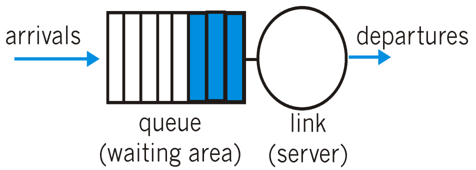


Principle 3

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

Scheduling And Policing Mechanisms

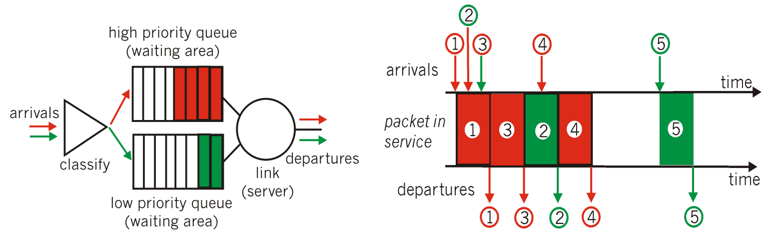
- scheduling: choose next packet to send on link
- FIFO (first in first out) scheduling: send in order of arrival to queue
  - ⇨real-world example?
  - discard policy: if packet arrives to full queue: who to discard?
    - Tail drop: drop arriving packet
    - priority: drop/remove on priority basis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



Scheduling Policies: more

Priority scheduling: transmit highest priority queued packet

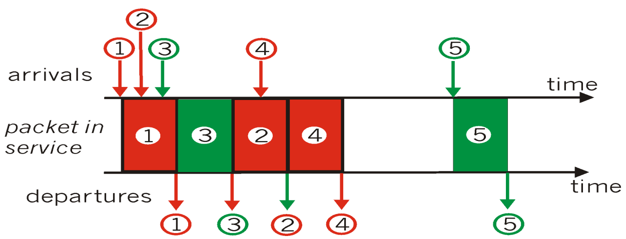
- multiple *classes*, with different priorities
  - class may depend on marking or other header info, e.g. IP source/dest, port numbers, etc..



Scheduling Policies: still more

round robin scheduling:

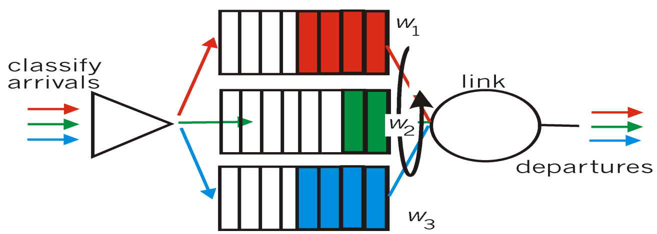
- multiple classes
- cyclically scan class queues, serving one from each class (if available)



Scheduling Policies: still more

Weighted Fair Queuing:

- generalized Round Robin
- each class gets weighted amount of service in each cycle
- when all classes have queued packets, class i will receive a bandwidht ratio of $w_i/\Sigma w_j$



Policing Mechanisms

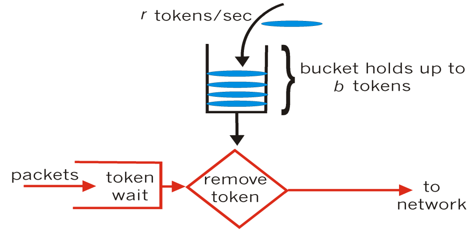
Goal: limit traffic to not exceed declared parameters

Three common-used criteria:

- *(Long term) Average Rate:* how many packets 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 packets per min. (ppm) avg.; 1500 pps peak rate
- *(Max.) Burst Size:* max. number of packets sent consecutively

Policing Mechanisms

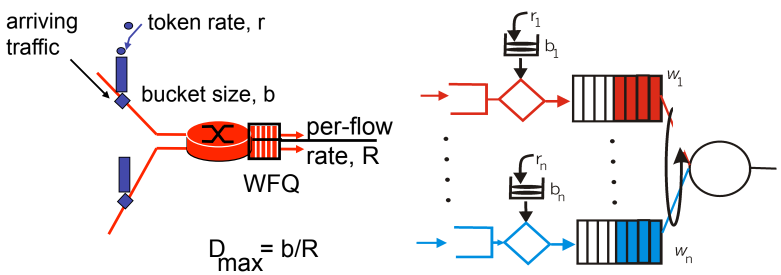
Token Bucket: limit input to specified Burst Size and Average Rate.



- bucket can hold b tokens ⇒ limits maximum burst size
- 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 combined provide guaranteed upper bound on delay, i.e., *QoS guarantee*



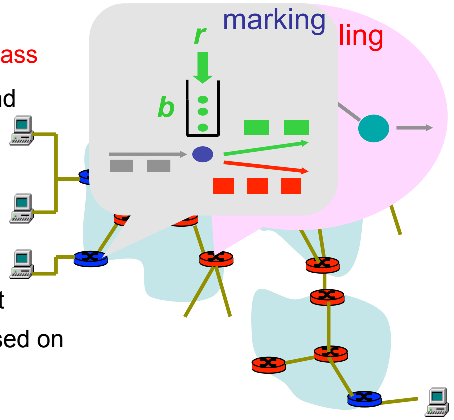
IETF Differentiated Services

- want "qualitative" service classes
  - "behaves like a wire"
  - relative service distinction: Platinum, Gold, Silver
- *scalability:* simple functions in network core, relatively complex functions at edge routers (or hosts)
  - in contrast to IETF Integrated Services: signaling, maintaining per-flow router state difficult with large number of flows
- don't define define service classes, provide functional components to build service classes

Diffserv Architecture

Edge router:

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

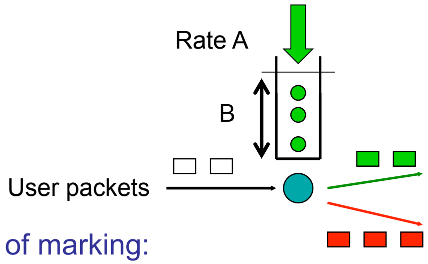


Core router:

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

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

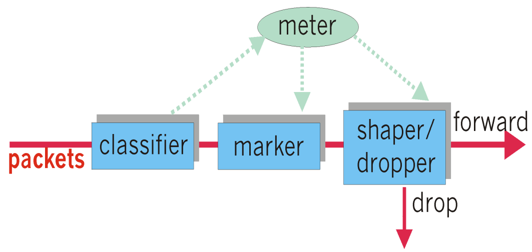
Classification and Conditioning

- 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 can be used for congestion notification: Explicit Congestion Notification (ECN), RFC 3168

Classification and Conditioning

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

- user declares traffic profile (e.g., rate, burst size)
- traffic metered, shaped or dropped 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:

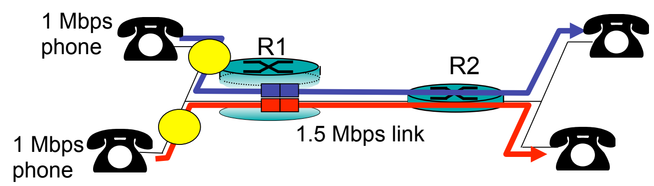
- Expedited Forwarding: packet departure rate of a class equals or exceeds specified rate
  - logical link with a minimum guaranteed rate
- Assured Forwarding: e.g. 4 classes of traffic
  - each class guaranteed minimum amount of bandwidth and a minimum of buffering
  - packets each class have one of three possible drop preferences; in case of congestion routers discard packets based on drop preference values

Chapter outline – Quality-of-Service Support

- Providing multiple classes of service
- Providing QoS guarantees
- **Signalling for QoS**

Principles for QOS Guarantees (more)

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

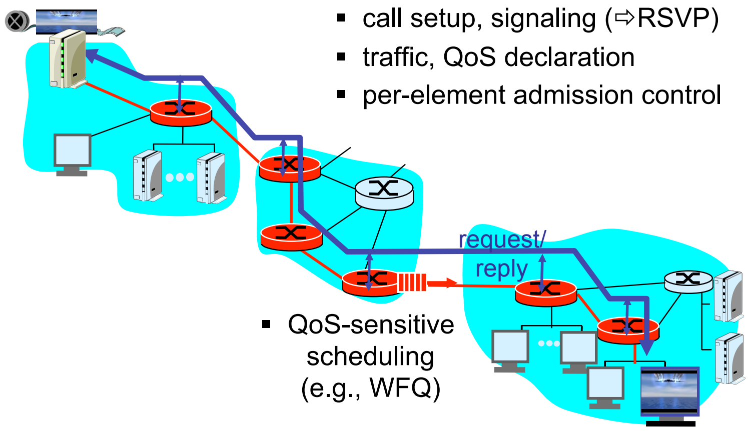


Principle

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

QoS Guarantee Scenario

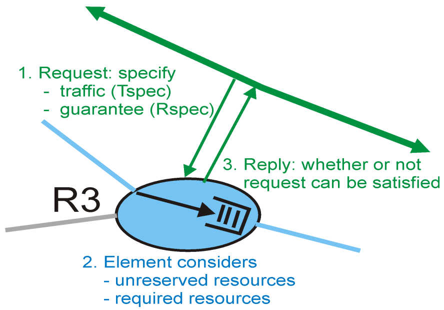
- Resource reservation
  - call setup, signaling (⇨RSVP)
  - traffic, QoS declaration
  - per-element admission control



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

Call Admission

- Routers will admit calls based on:
- Flow behavior:
  - R-spec and T-spec
- the current resource allocated at the router to other calls.



IETF Integrated Services

- architecture for providing QOS guarantees in IP networks for individual application sessions
- resource reservation: routers maintain state info (as for VCs) of allocated resources, QoS requests
- 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?

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

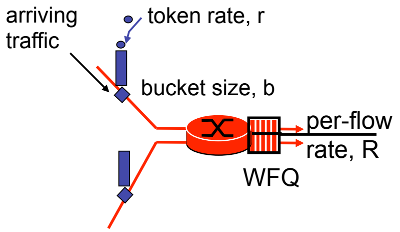
Intserv QoS: Service models [RFC 2211, RFC 2212]

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."



$D_{max} = b/R$