

Quality of Service in Internet

- ✧ Some materials come from Cisco.com and some from Computer Networking: A Top Down Approach, 6th edition, Jim Kurose, Keith Ross, Addison-Wesley March 2012

Outline

- ***Introduction***
- **Integrated Services (IntServ)**
- **Differentiated Services (DiffServ)**

What Is QoS?

The Application/User Perspective

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

What Is QoS?

The Network Perspective

- **The definition of a PIPE:**
 - The path from point A to point B, as **perceived** by a Packet
 - Similar to your experience in driving from city A to city B!
- **QoS is the set of techniques to manage:**
 - Bandwidth/Throughput—the perceived width of the Pipe
 - Delay—the perceived length of the Pipe
 - Jitter—the perceived variation in the length
 - Packet Loss—the perceived leak in the Pipe

What Is QoS?

The Network Perspective (Cont.)



Motivation of providing QoS in Internet

- Internet gives “best-effort” delivery service to all
- Treats all packets equally, with no service levels, requirements, reservations, guarantees or admission control
- Real-time, multimedia, multicasting (inelastic traffic) applications are not well supported
- Support variety of traffic with a variety of QoS requirements within TCP/IP
- Translates a request for service into a set of requirement (throughput, delay, jitter, loss, error rates)

Internet Traffic Types

1. Elastic Traffic

- Applications are flexible over the required delay and throughput (faster is better)
- e.g. file transfer (FTP); email (SMTP); remote logon (Telnet); network management (SNMP); web (HTTP)

2. Inelastic Traffic

- Applications are intolerant to change in delay and throughput (a min value required), jitter (variation of delay) and packet loss.
- e.g. real-time traffic such as interactive voice and video

QoS Architecture Analogy...

- **Think of your postal system!**
 - **A Packet ~ letter/package**
 - **A Router ~ mail-hub/Post-Office**
 - **A Link ~ land/sea/air route**

Evolution of the QoS Architectures...

- **Best-Effort IP Service: 1981, RFC-791**
 - Compare to regular mail service
- **Integrated Services (IntServ)/RSVP: 1997**
 - Compare to a custom mail service
 - (e.g. DHL, FedEx or UPS)
- **Differentiated Services (DiffServ): 1998/99**
 - Compare to mail service classes
 - (Express, Priority, Registered, Regular, etc.)

IntServ Vs. DiffServ

- **IntServ:**
 - **QoS on a per-flow basis**
 - **Developed for ATM networks**
- **DiffServ**
 - **QoS for flow aggregates, e.g. all the flows belonging to the same characteristics are grouped together as a larger flow or a class**

Outline

- Introduction
- ***Integrated Services (IntServ)***
- Differentiated Services (DiffServ)

IntServ

- **Focus on QoS of IP packets between two end points; each flow requests a specific level of service (e.g. min rate, max delay or loss)**
- **Resource reservations are established and maintained on a per-flow basis**

Major components:

- 1. Admission control**
- 2. Packet forwarding mechanism**
- 3. Resource Reservation Protocol (RSVP) - used to set up resources in the routers along intended path**

Reminder: IntServ is not widely accepted due to scalability and manageability problems

Outline

- Introduction
- Integrated Services (IntServ)
- ***Differentiated Services (DiffServ)***

DiffServ

- **Attempt to overcome problems in scalability and manageability**
- **Classify and regulate traffic at the boundaries of network**
- **Focus on flows of aggregated traffic in classes according to specific characteristics**
- **Label packets with a field (hence more scalable and manageable architecture)**
- **Use the type field for QoS management in the core of the network (hence simple)**

Diffserv architecture

edge router: 

- ❖ *per-flow* traffic management

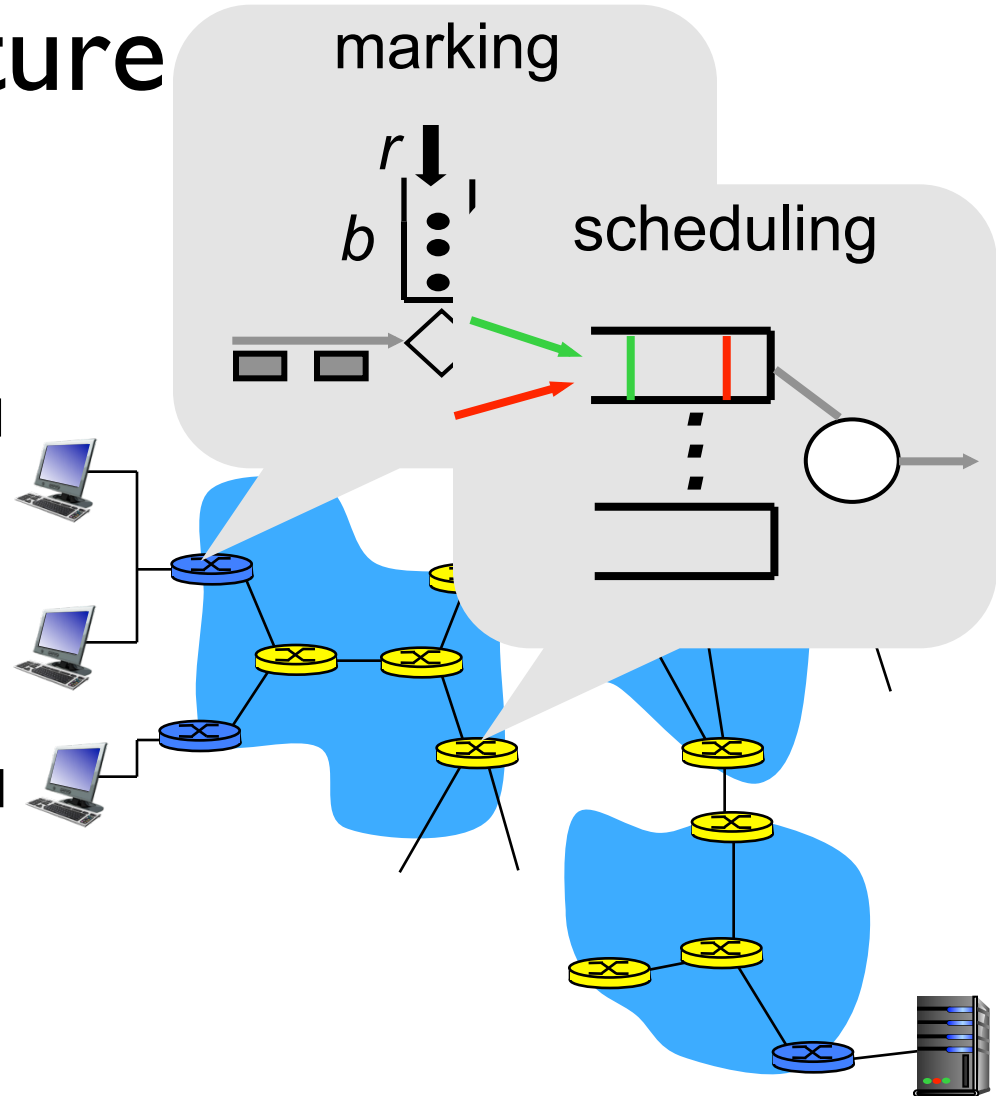
- ❖ marks packets as **in-profile** and **out-profile**

core router: 

- ❖ *per class* traffic management

- ❖ buffering and scheduling based on *marking* at edge

- ❖ preference given to **in-profile** packets over **out-of-profile** packets

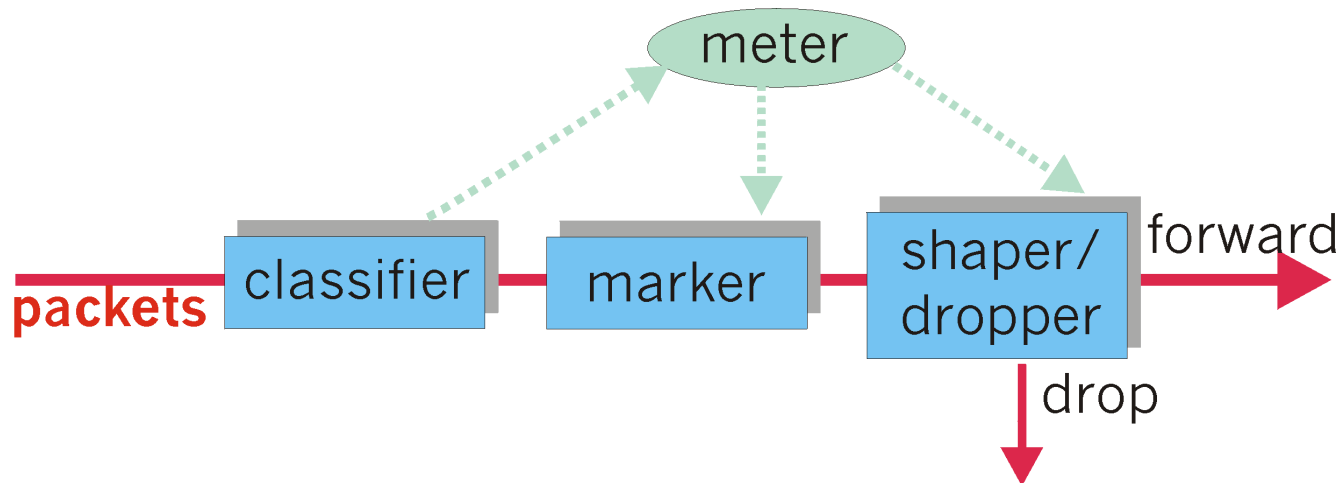


The DiffServ Model

- The idea is **VERY** simple—Offer service levels for packets: **Platinum/Premium, Gold, Silver and Bronze**
- Packets of a particular service are referred to as packets of a particular “class”
- Meaningful services constructed using **Per-Hop Behaviors (PHB)**

DiffServ-Traffic Conditioner Block (TCB)

- **Classifier:** Identifies packets for assignment to Classes
- **Meter:** Checks compliance to traffic parameters and passes result to Marker and Shaper/Dropper to trigger particular actions for out-of-profile packets (e.g. remark, delay and drop packets)
- **Marker:** Writes/rewrites the DiffServ Code Point (DSCP) value, indicating Type of Service
- **Shaper:** Delays some packets for them to be compliant with the profile
- **Dropper:** Drops packets that exceed the profile



DiffServ Recipe for Constructing Services

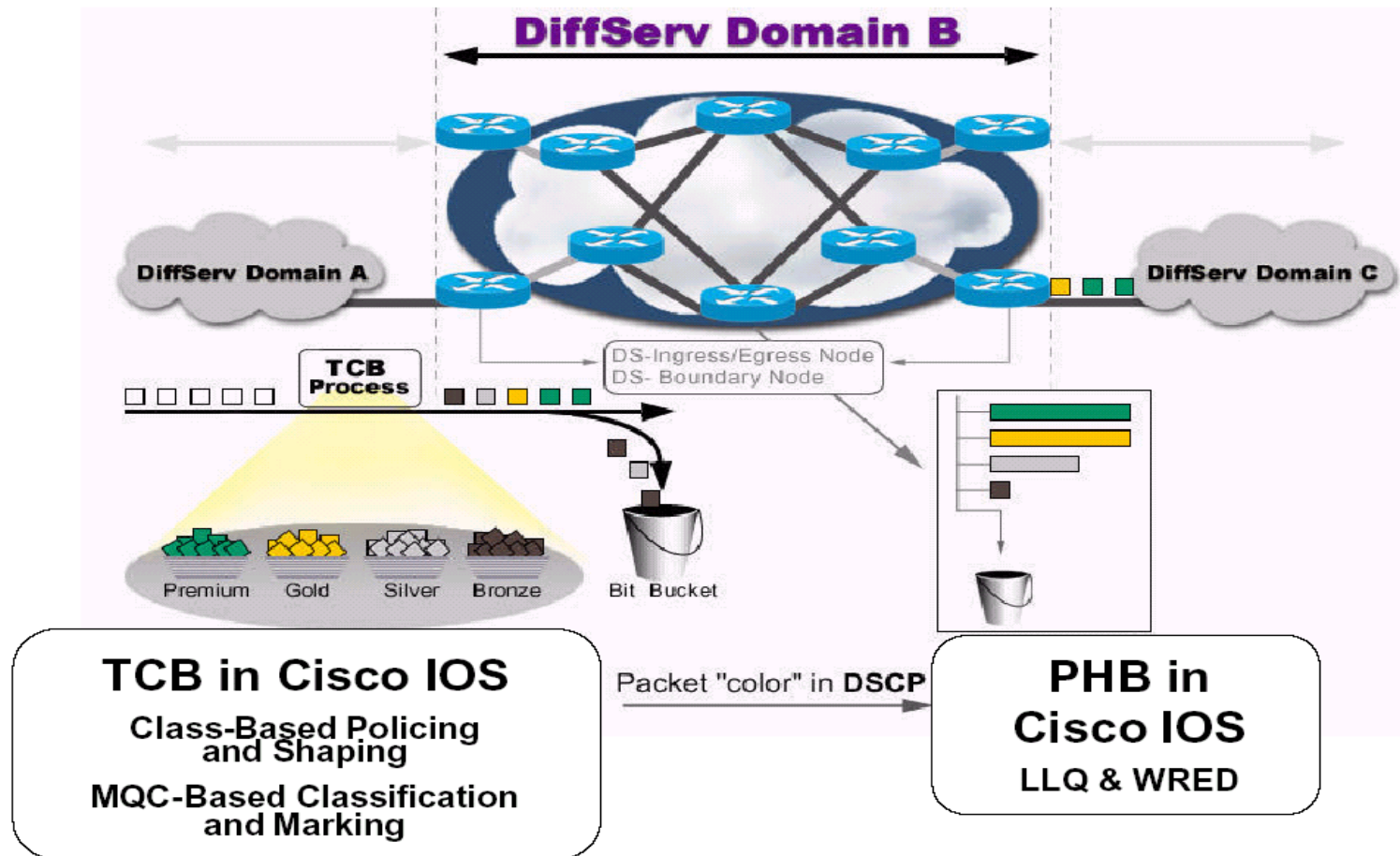
At the Ingress Network-Edge:

- 1) Classify the packets into 'Classes'
- 2) Mark (Color) the packets for purposes of classification in the core
- 3) Meter a class
- 4) Remark (recolor) out-of-profile packets for the class
- 5) Delay and/or drop out-of-profile packets toward the core

In the network core: (implementing the PHB)

- 6) Queue and/or drop packets

The DiffServ Architecture (RFC-2475)

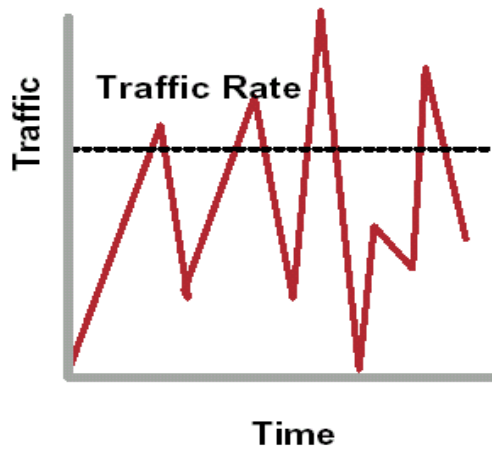


Packet Classification

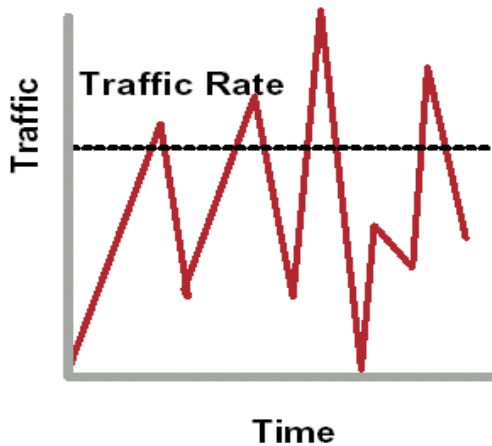
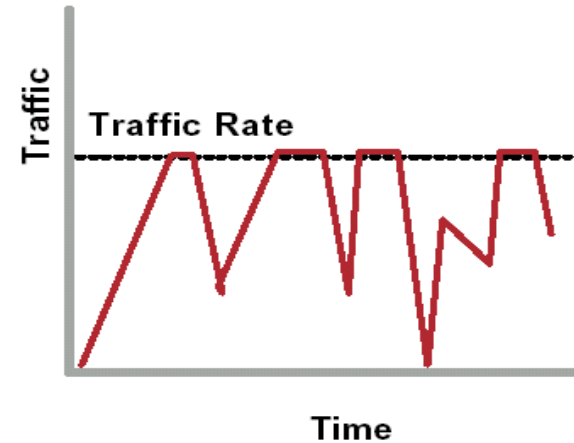


- **Packet is marked in the Type of Service (TOS) in IPv4, and Traffic Class in IPv6.**
- **6 bits used for DiffServ Code Point (DSCP) 2 bits are currently unused.**

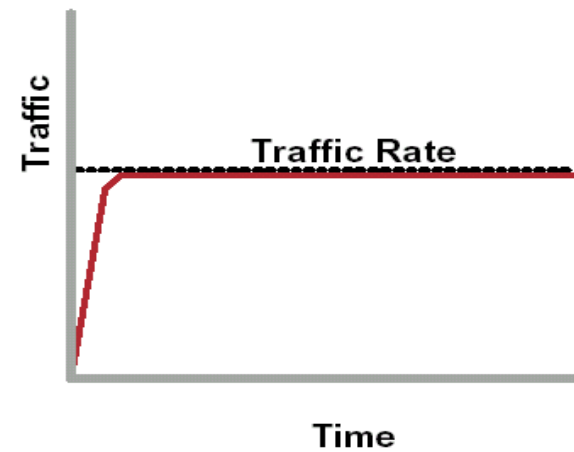
Metering $\rightarrow$ Policing (Dropping)/Shaping



Policing

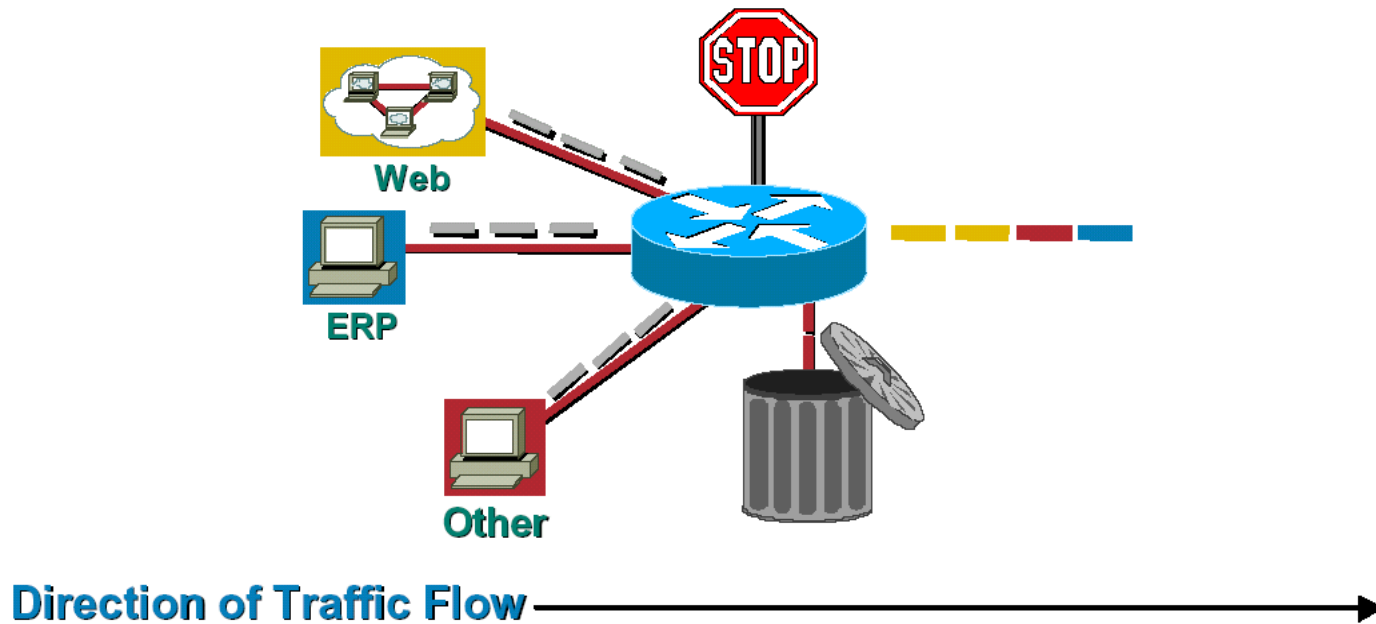


Shaping



More on Policing...

- Policing is used not only to drop out-of-profile packets, but also to **re-mark (re-color)** them, and indicate to dropping mechanisms downstream that they should be dropped ahead of the in-profile packets!



Policing mechanisms

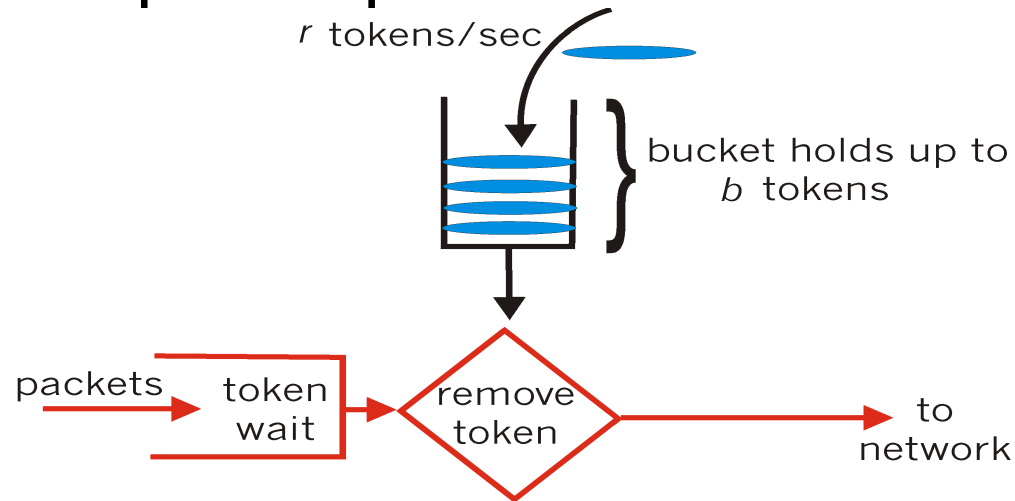
goal: limit traffic to not exceed declared parameters

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.; 15000 ppm peak rate
- **(max.) burst size:** max number of pkts sent consecutively (with no intervening idle)

Policing mechanisms: implementation

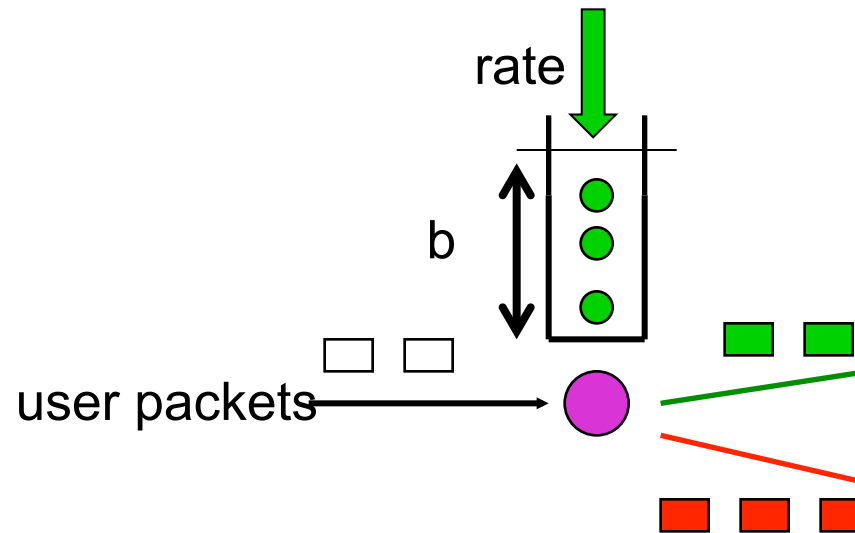
token bucket: limit input to specified **burst size** and **average rate**



- 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 $(r t + b)$*

Edge-router packet marking

- ❖ *profile*: pre-negotiated rate r , bucket size b
- ❖ packet marking at edge based on *per-flow* profile



possible use of marking:

- ❖ class-based marking: packets of different classes marked differently
- ❖ intra-class marking: marks packets as **in-profile** and **out-profile**

The Various PHBs (Using Queuing and Dropping)

- **Expedited Forwarding (EF): RFC2598**
 - Very low delay, low jitter, assured bandwidth
 - Compare to express mail: with overnight delivery
- **Assured Forwarding (AF): RFC2597**
 - Assured some amount of bandwidth
 - IETF has defined four AF classes and three drop priorities for each AF class
 - Compare to registered mail: very safe, and assured
- **Default: Best effort ~ normal mail**

The DiffServ Architecture Again...

