Quality of Service (QoS) Primer

Avinash Tadimalla

Agenda Du Jour

- **What is QoS?**
- Why is it Required?
- QoS Mechanisms
- QoS Architectures
- QoS Deployment Guide
- Q and A (and C)

What is QoS?

- Quality of Service is an attempt to provide **predictable response** for applications from end-point to end-point by administratively applying different services within the network infrastructure for the applications

OR

- Quality of Service refers to the capability of a network to provide **better service** to selected network traffic

OR

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

What is QoS (contd)

- "(Better) performance" as described by a set of parameters or measured by a set of metrics.
- Generic parameters:
  - Bandwidth
  - Delay, Delay-jitter
  - Packet loss rate (or probability)
- Transport/Application-specific parameters:
  - Timeouts
  - Percentage of "important" packets lost
- These parameters can be measured at several granularities:
  - "micro" flow, aggregate flow, population.
- QoS considered "better" if
  - a) more parameters can be specified
  - b) QoS can be specified at a fine-granularity.

What is QoS - Three Perspectives

The user perspective

- **Users perceive that their applications are performing properly**

  Voice, video, and data

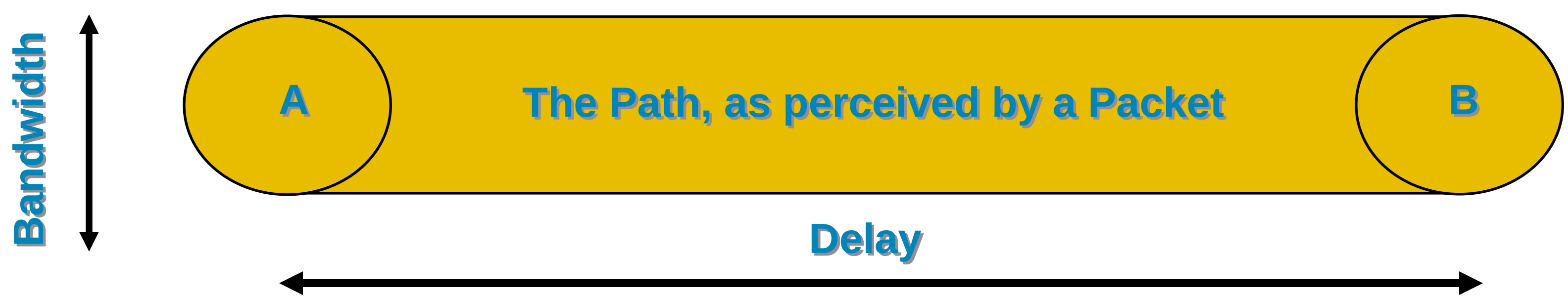
The network manager perspective

- **Need to manage bandwidth allocations to deliver the desired application performance**

  Control delay, jitter, and packet loss

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—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

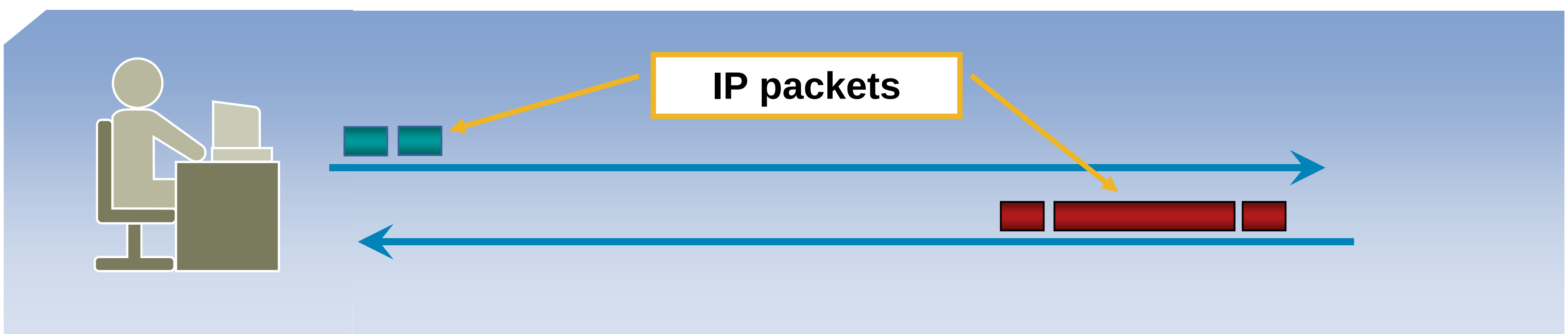


Agenda Du Jour

- What is QoS?
- **Why is it Required?**
- QoS Mechanisms
- QoS Architectures
- Summary
- Q and A (and C)

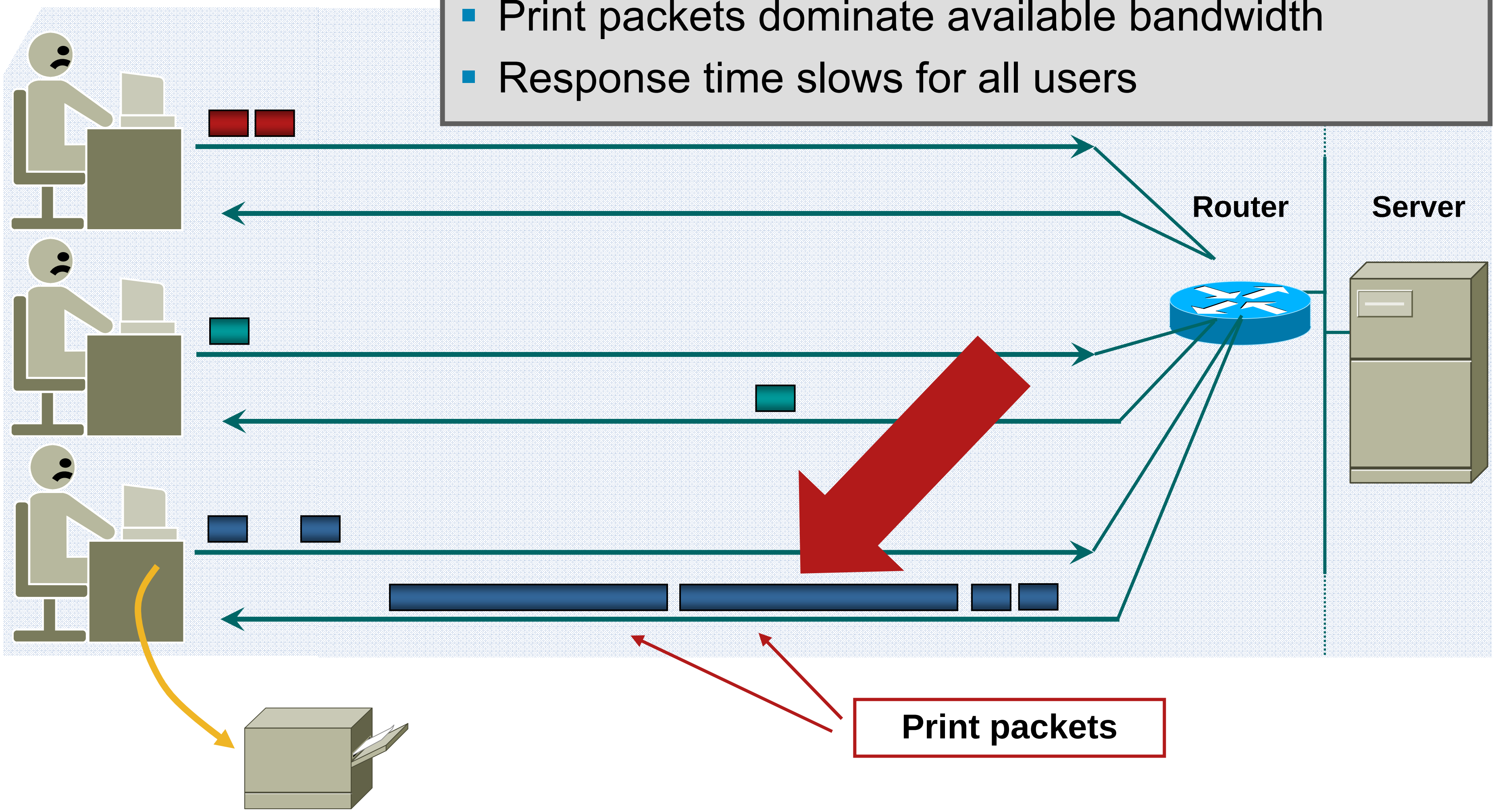
"Best Effort" Quality of Service

- Without QoS policies, traffic is served with "best effort"
  - **No distinction between high and low priority**
    - Business critical vs. background
  - **No allowances for different application needs**
    - Real-time voice/video vs. bulk data transfer
- No problem, until congestion occurs



Congestion without QoS – Example: User Prints to Attached Printer

- Print packets dominate available bandwidth
- Response time slows for all users

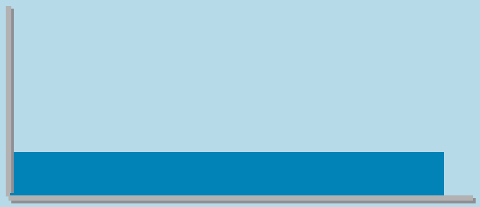


Why better-than-best-effort?

- To support a wider range of applications with unpredictable demands
  - Real-time, Multimedia etc
  - Minimize Packet Loss, Delay and Delay Variation/Jitter
- To develop sustainable economic models and new private networking services
  - Current flat priced models, and best-effort services do not cut it for businesses
- Offer Differentiated Services for Profitability:
  - Premium-Class Service (VoIP, Stock Quotes)
  - Business-Class Service (SAP, Oracle, Citrix)
  - Best-Effort Service – (Backups, Email)

QoS Requirements for Voice

Voice



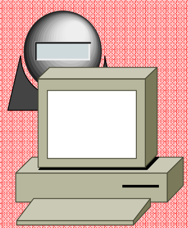
- Smooth
- Benign
- Drop Sensitive
- Delay Sensitive
- UDP Priority

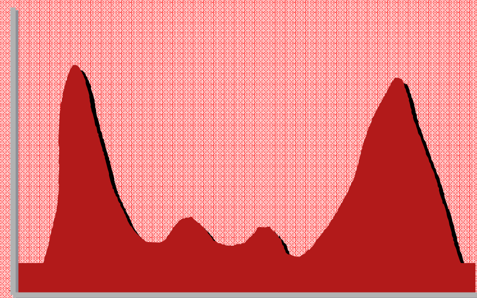
- Latency ≤ 150 ms
- Jitter ≤ 30 ms
- Loss ≤ 1%
- 17-106 kbps guaranteed priority bandwidth per call
- 150 bps (+ layer 2 overhead) guaranteed bandwidth for Voice-Control trafffic per call

One-way requirements

QoS Requirements for Video-Conferencing

Video





- Bursty
- Greedy
- Drop Sensitive
- Delay Sensitive
- UDP Priority

- Latency ≤ 150 ms
- Jitter ≤ 30 ms
- Loss ≤ 1%

One-way requirements

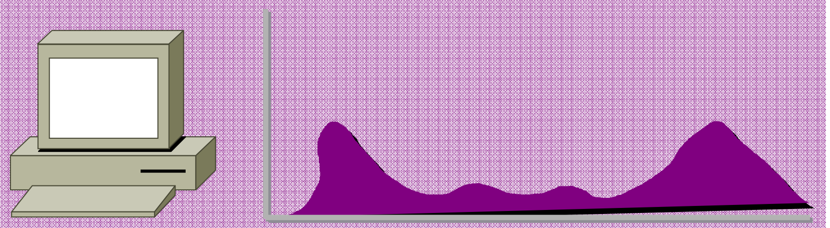
- Minimum priority bandwidth guarantee required is:

  Video-Stream + 20%

  e.g. a 384 kbps stream would require 460 kbps of priority bandwidth

QoS Requirements for Data

Data



- Smooth/Bursty
- Benign/Greedy
- Drop Insensitive
- Delay Insensitive
- TCP Retransmits

- Different applications have different traffic characteristics
- Different *versions* of the same application can have different traffic characteristics
- Classify Data into relative-priority model with no more than four classes:

  Gold: Mission-Critical Apps
  (ERP Apps, Transactions)

  Silver: Guaranteed-Bandwidth
  (Intranet, Messaging)

  Bronze: Best-Effort
  (Internet, Email)

  Less-Than-Best-Effort: Scavenger
  (FTP, Backups, Napster/Kazaa)

Why is QoS required - Recap

- Some congestion is likely in most networks
- Over-provisioning is NOT the solution
- Provides the ability to control transmission quality of the network under congestion
- Transmission quality
  - Latency
  - Throughput
  - Jitter
  - Packet Loss
- Different applications are sensitive to different characteristics
- Always good to carry an “insurance” policy

Agenda Du Jour

- What is QoS?
- Why is it Required?
- QoS Mechanisms
- QoS Architectures
- QoS Deployment Guide
- Q and A (and C)

The Building Blocks

- a) Specification of premium services

  (service/service level agreement design)

- b) How much resources to set aside?

  (admission control/provisioning)

- c) How to ensure network resource utilization, do load balancing, flexibly manage traffic aggregates and paths ?

  (QoS routing, traffic engineering)

- d) How to actually set aside these resources in a distributed manner ?

  (signaling, provisioning, policy)

- e) How to deliver the service when the traffic actually comes in (claim/police resources)?

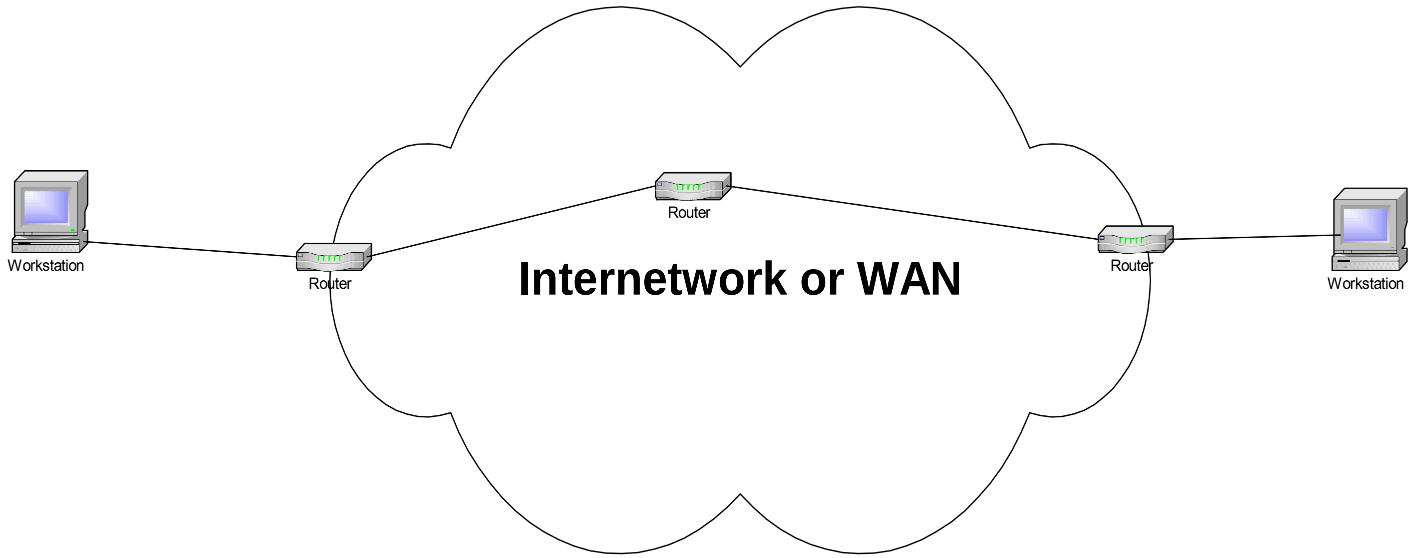
  (traffic shaping, classification, scheduling)

- f) How to monitor quality, account and price these services?

  (network mgmt, accounting, billing, pricing)

The Big Picture – Control vs. Data Planes

Control Plane: Signaling + Admission Control **or** SLA (Contracting) + Provisioning/Traffic Engineering



Data Plane: Traffic conditioning (shaping, policing, marking etc) + Traffic Classification + Scheduling, Buffer management

QoS Mechanisms - Classification

- What is Classification

  The most fundamental component of QOS

  Classification is the process of identifying traffic and categorizing it into different classes.

  The goal is to identify packets in order to match them to their QoS requirements

- Classification Tools

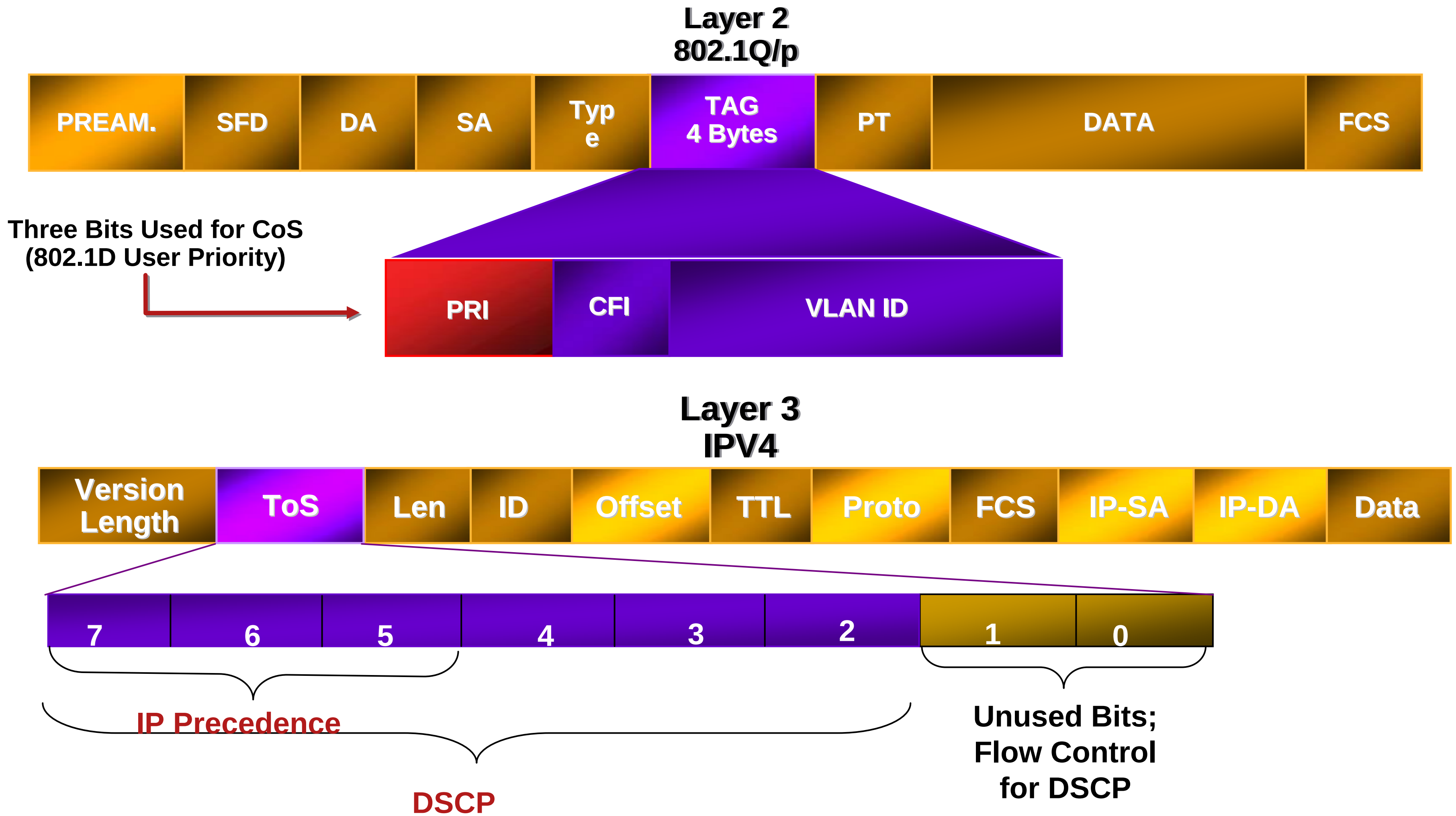
  Access Lists : layers 2-4 classification engine

  Protocol Based (NBAR) : layer 5-7 classification engine

- Enforce a Trust Boundary

  Classification and Trust Boundaries as close to the edge as possible

Identifying Traffic - Layer 2 and 3

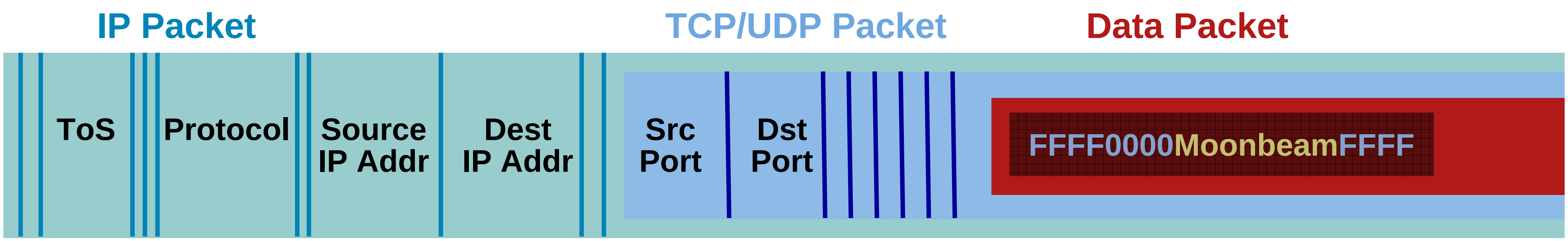


First 3 MSBs define the IP Precedence or Type of Service
(DiffServ May Use Six D.S. Bits Plus Two for Flow Control)

Network-Based Application Recognition (NBAR)

- IP packet classifier capable of classifying applications that have:
  - Statically assigned TCP and UDP port numbers
  - Non-TCP and non-UDP IP protocols
  - Dynamically assigned TCP and UDP port numbers during connection establishment
  - Classification based on deep packet inspection—NBAR's ability to look deeper into the packet to identify applications
- Currently supports over 100 protocols/applications

NBAR User-Defined Custom Application Classification



- Name—Name the match criteria—up to 24 characters
  my_protocol
- Offset—Specify the beginning byte of string or value to be matched in the data packet, counting from ZERO for the first byte
  Skip first 8 bytes
- Format—Define the format of the match criteria—ASCII, hex or decimal
  ascii
- Value—The value to match in the packet—if ASCII, up to 16 characters
  Moonbeam
- [Source or destination port]—Optionally restrict the direction of packet inspection; defaults to both directions if not specified
  [source | destination]
- TCP or UDP—Indicate the protocol encapsulated in the IP packet
  tcp
- Range *or* selected port number(s)
  "range" with start and end port numbers, up to 1000
  1 to 16 individual port numbers
  range 2000 2999

Example

ip nbar custom my_protocol
8 ascii Moonbeam tcp
range 2000 2999

class-map custom_protocol

match protocol my_protocol

policy-map my_policy
class custom_protocol
set ip dscp AF21

interface <>

service-policy output my_policy

QoS Mechanisms - Marking

- What is Marking

  The QOS component that "colors" a packet (frame) so it can be identified and distinguished from other packets (frames) in QOS treatment

  Once the packet is classified into a specific service class, marking the packet header allows the core networking elements to apply the appropriate QoS technologies to the packet in an efficient manner

- Marking Tools

  Class of Service (ISL, 802.1p)

  IP Precedence

  DSCP

  PHB

Marking Techniques

- There exist multiple packet marking techniques including:

  Layer 3: IPv4 IP Precedence Field
  IPv4 DiffServ Differentiated Services Field
  IPv6 DiffServ Differentiated Services Field

  Layer 2: MPLS Exp/CoS Field
  802.1d (802.1p+q) User Priority Field
  ISL User Priority Field

- Layer 2 versus Layer 3 Marking

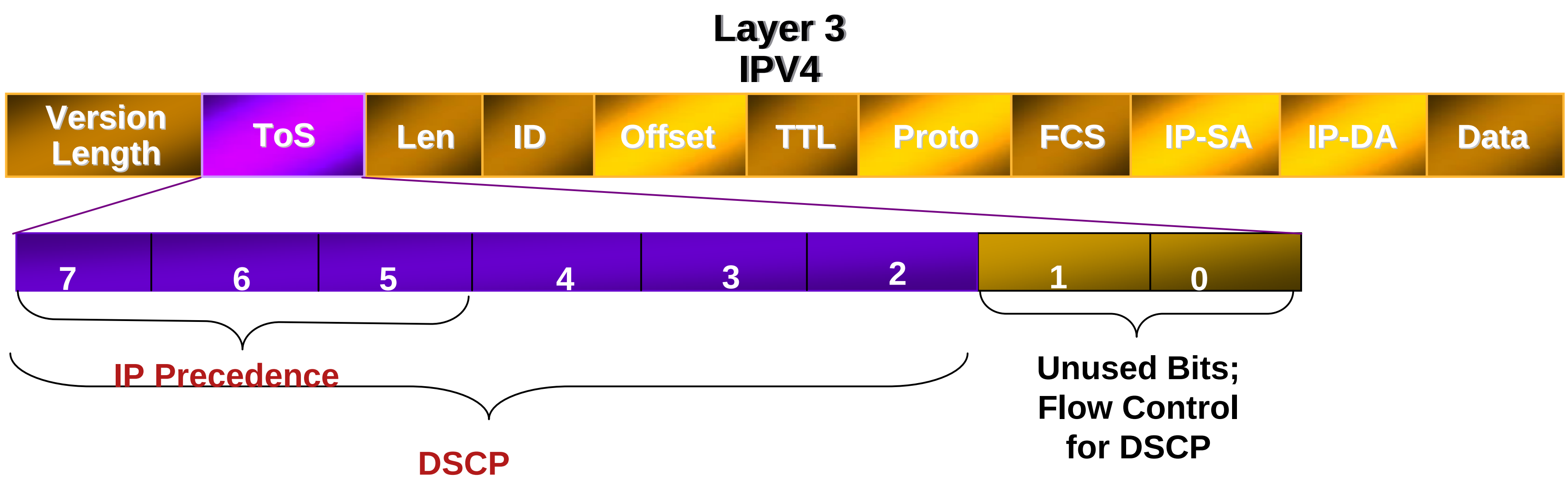
  Layer 2 Ethernet Class of Service (CoS) settings (802.1q Header)

  Three bits allow for 7 levels of classification

  These levels directly correspond to IPv4 ToS values

  However, Layer 3 marking is more ubiquitous (Why?)

IP Precedence and DiffServ Code Points



First 3 MSBs define the IP Precedence or Type of Service
(DiffServ May Use Six D.S. Bits Plus Two for Flow Control)

- IPv4: Three Most Significant Bits of ToS byte are called IP Precedence (IPP); other bits unused
- DiffServ: Six Most Significant Bits of ToS byte are called DiffServ Code Point (DSCP); remaining two bits used for flow control
- DSCP is backward-compatible with IP Precedence; an instance of DSCP is a Per Hop Behavior (PHB)

QoS Mechanisms - Congestion

- What is Congestion

  When the offered load exceeds the capacity of a data communication path, the resulting situation is called Congestion.

  Congestion can occur at any point in the network where there are speed mismatches or link aggregations

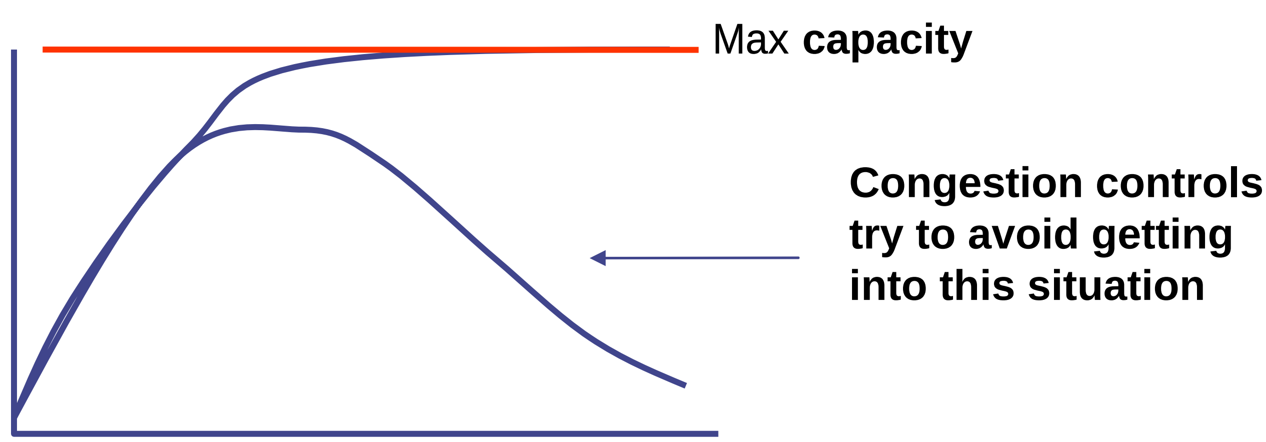
- Congestion Tools

  Congestion Management : is done by queuing packets

  Congestion Avoidance : is done by dropping packets

The Impact of Congestion

- Packet queues at links start to grow…
- Packets start dropping
- Sources start re-transmitting
- After a while only re-transmissions occupy the network
- Network resources start getting utilized in useless work (packets in queues that get timed out and re-transmitted)
- “Goodput” goes to nearly zero



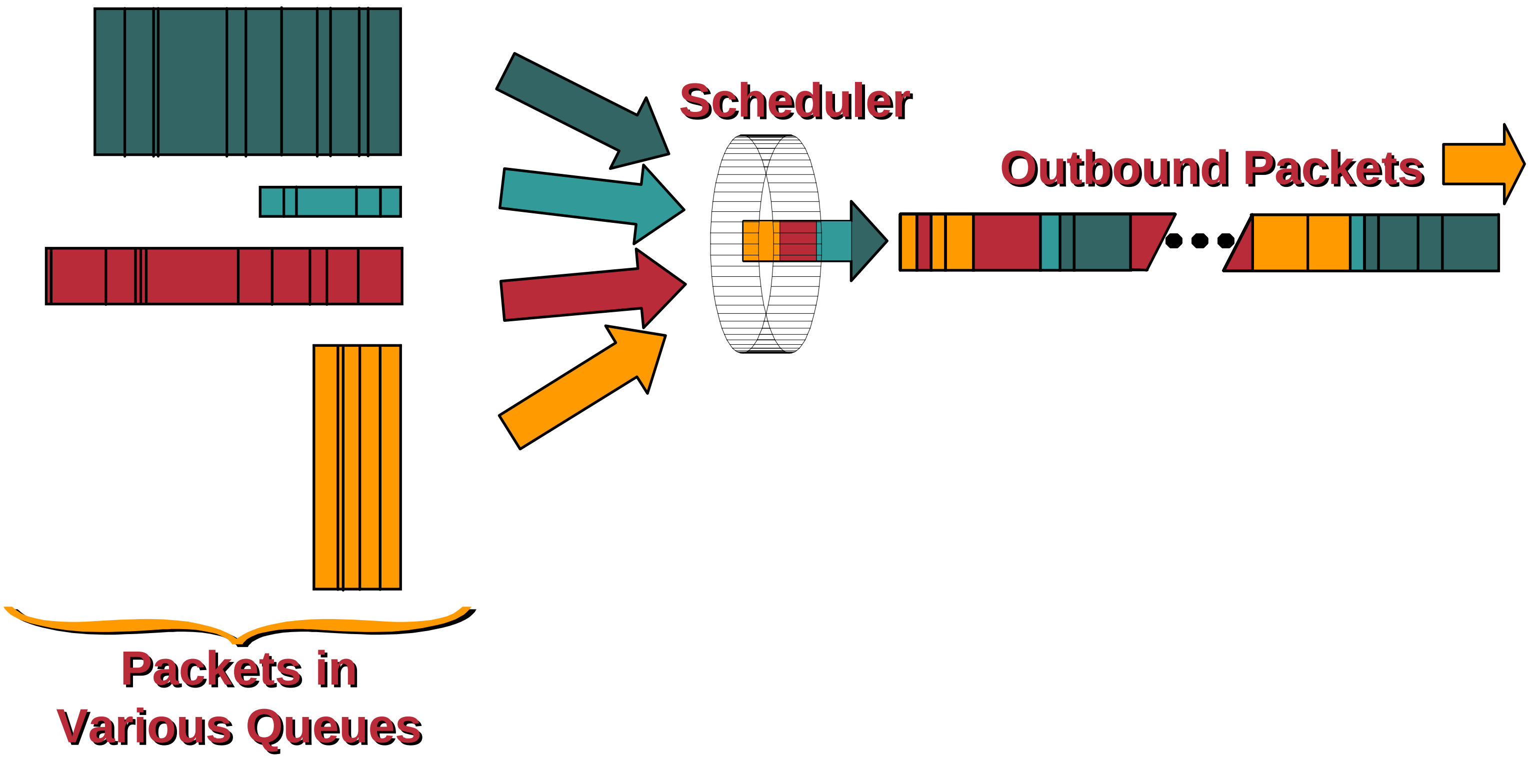
Congestion Management

- Is done by Queuing
- Queuing algorithms manage the front (scheduling) of a queue
- These algorithms control
  - the order in which the packets are sent
  - the usage of the router's buffer space
- Queuing Algorithms:

First In First Out (FIFO)	**Legacy Congestion management**
Priority Queuing (PQ)	**Legacy Congestion management**
Custom Queuing (CQ)	**Legacy Congestion management**
Weighted Fair Queuing (WFQ)	**Legacy Congestion management**
Class-Based WFQ (CBWFQ)	**Latest Congestion management**
PQ-CBWFQ (LLQ)	**Latest Congestion management**
PQ-WFQ (IP RTP Priority)	**Latest Congestion management**

Congestion Management – Graphical View

- Buffers packets when interface is congested
- Schedules packets out of the buffer onto the link (Algorithms: FIFO, CBQ, etc.)

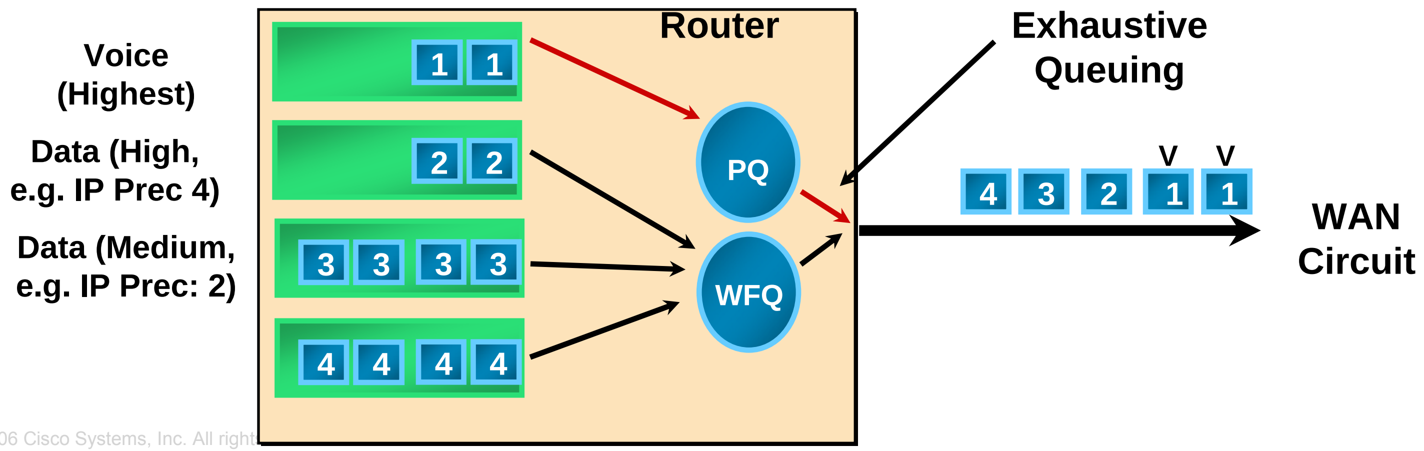


Queuing Algorithms – Class Based Weighted Fair Queuing (CBWFQ)

- Combines the capability to guarantee bandwidth (from CQ) with the capability to dynamically ensure fairness to other flows within a class of traffic (from WFQ)
- In WFQ, bandwidth allocations change continuously, as flows are added/ended
- CBWFQ adds a level of administrator control to the WFQ process; administrator can control how packets are classified
- BUT
  - **No latency guarantees**
  - **Human analysis / configuration**

Queuing Algorithms – Priority Queuing-WFQ (PQ-WFQ)

- Also known as IP RTP Priority Queuing
- To prioritise Voice traffic (on FR, PPP)
- Create a priority queue (weight=0) + BW limit
- Essentially gives the router two WFQ systems, one for normal traffic and another for voice
- voice is serviced as strict priority in preference to other non-voice traffic.
- RTP only (range of UDP ports)



Queuing Algorithms – Low Latency Queuing (LLQ)

- Also known as Priority Queuing - CBWFQ
- Provides a single priority queue, like PQ-WFQ
- Guaranteed bandwidth for different traffic classes can be configured
- LLQ Specifies **maximum bandwidth** in Kbps that a flow is assured under congestion as opposed to the minimum bandwidth guaranteed by CBWFQ
- Multiple priority classes are all enqueued in a single priority queue but policed and rate limited individually
- Guarantees Bandwidth and Restrains flow of packets from priority class ensuring non priority packets are not bandwidth starved

Queuing Algorithms – Recap

- Newer queuing algorithms are hybrid combinations of basic queuing algorithms
- IP RTP Priority

  intermediate solution (until LLQ developed) to assign voice PQ, but without starving data (which received WFQ)
- CBWFQ

  combination of CQ and WFQ

  minimum bandwidth guarantees can be made, but also WFQ can take place within classes

  very efficient algorithm for data applications
- LLQ

  combination of PQ, CQ and WFQ

  PQ-like treatment of voice and/or video

  efficient handling of data traffic with minimum bandwidth guarantees

Congestion Avoidance

- Congestion avoidance mechanisms are complementary to (and dependant on) queuing algorithms.
- Queuing algorithms manage the front of a queue, while congestion avoidance mechanisms manage the tail of the queue.
- Congestion Avoidance Tools

    Tail Drop

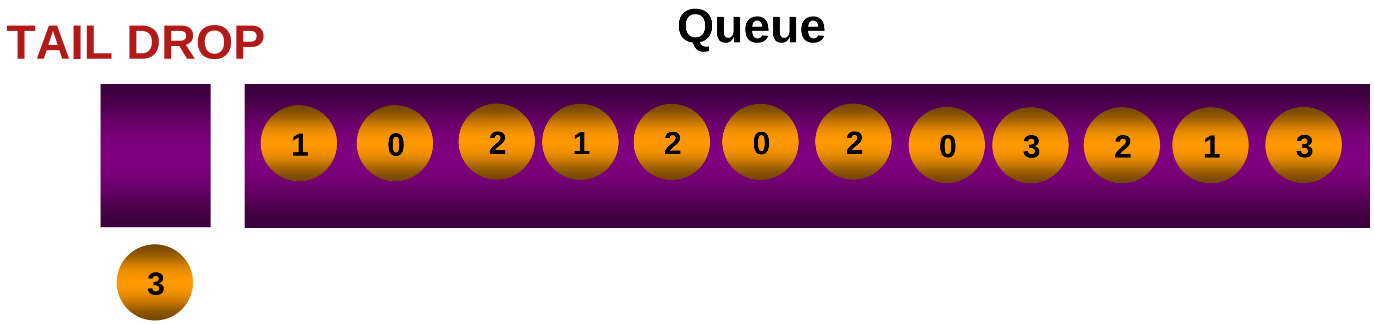
    RED

    WRED

The Need for Congestion Avoidance: Active Queue Management (AQM)

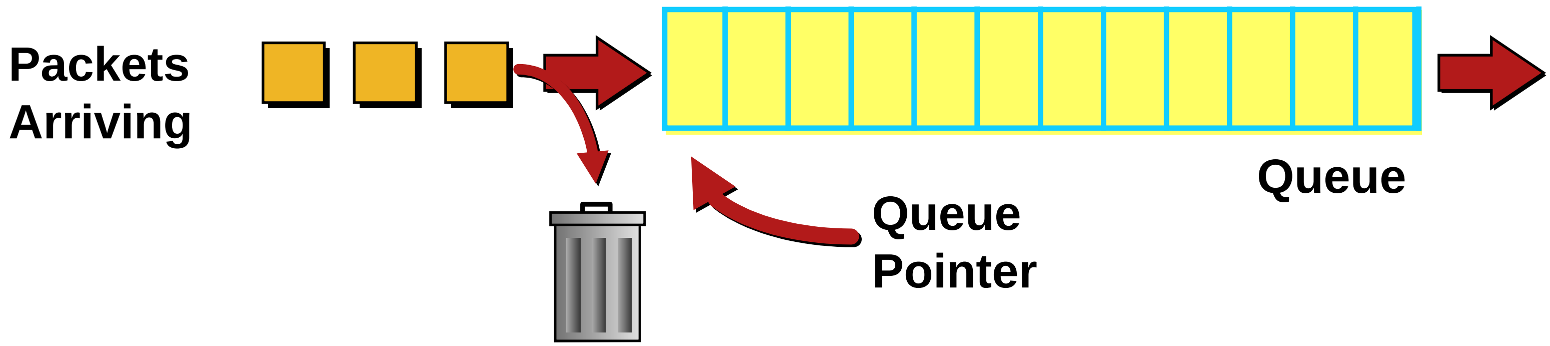
- Dropping can occur in the edge or core due to policing or buffer exhaustion
- If a queue fills up, all packets at tail end of queue get dropped—called tail-drop
- Tail-drop results in simultaneous TCP window shrinkage of large number of sessions, resulting in “global synchronization”
- Manage queue lengths by dropping packets when congestion is building up
- Works best with TCP-based applications, as selective dropping of packets causes the TCP windowing mechanisms to 'throttle-back' and adjust the rate of flows to manageable rates.

Congestion Avoidance – Tail Drop



- "Tail drops" occur when the transmit queue fills up and there is no room left for additional packets
- Without any type of congestion avoidance algorithm the higher priority packet (IP Prec 3) gets dropped –bad!

Congestion Avoidance – Random Early Detection (RED)



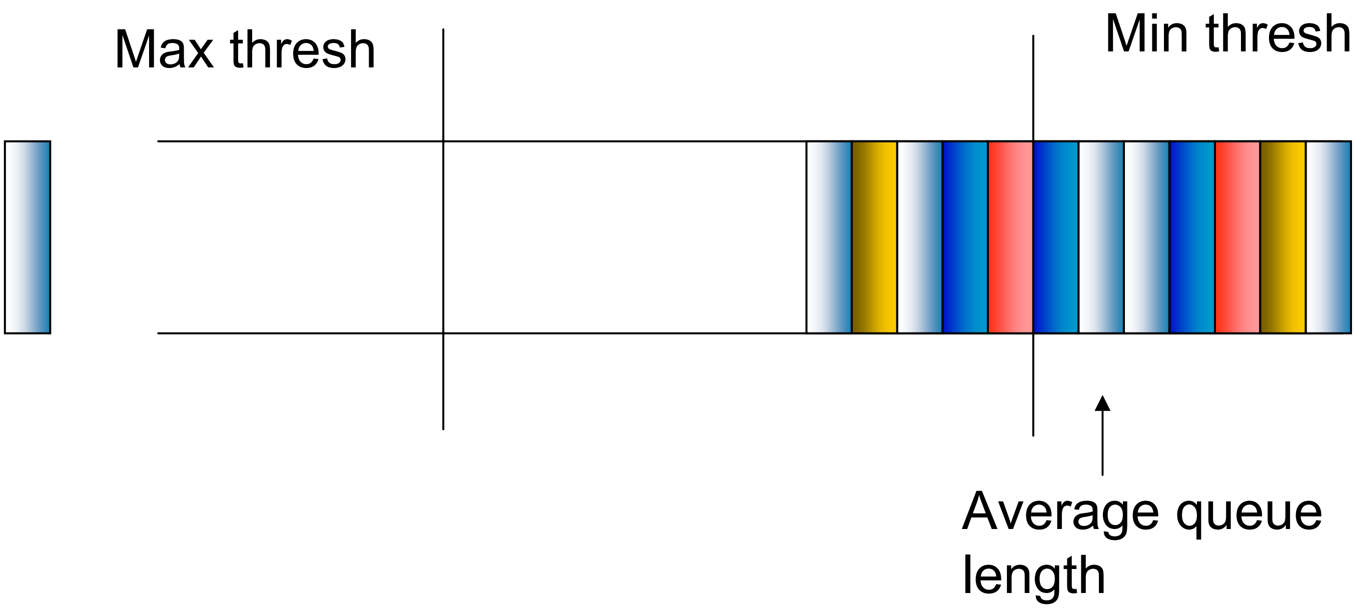
- The basic RED mechanism is to randomly drop packets before the buffer is completely full
- Depending on the average queue length, the drop probability is calculated

RED – Functional Description

- When a packet arrives, the following events occur:
  - The average queue size is calculated
  - If the average is less than the minimum queue threshold, the arriving packet is queued
  - If the average is between the minimum queue threshold and the maximum threshold, the packet is either dropped or queued, depending on the packet drop probability
  - If the average queue size is greater than the maximum threshold, the packet is automatically dropped

RED – Functional Description (Contd.)

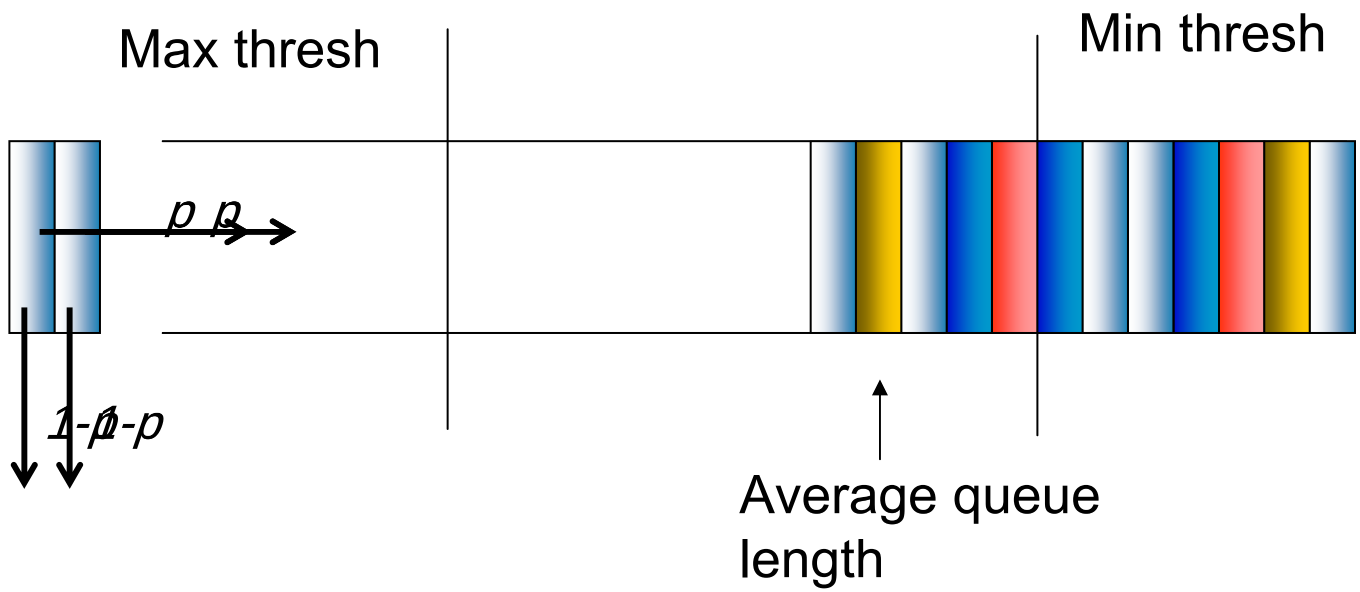
Case 1:
Average Queue Length < Min. Thresh Value



Admit the New Packet

RED – Functional Description (Contd.)

Case 2: Average Queue Length between Min. and Max. Threshold Value



Or Drop the New Packet With Probability 1-*p*

Advantages of RED

- Goal of congestion avoidance by controlling of average queue length
- The time scale from marking of packet to actual reduction in arriving packets is set appropriately
- Avoidance of global synchronization achieved by

  Randomness: by randomly choosing which packets to drop we do not drop all packets at the same time, hence causing all flows to back off in synchronously

  Low-drop rate: RED begins to drop as soon as min. threshold is exceeded, and the first levels of drops are pretty low so that only a few flows (statistically the more bandwidth demanding flows) will get dropped and obliged to back off.
- The proportion of marked packets in a connection is relative to its bandwidth share

Drawbacks of RED

- Packet loss rate independent of the bandwidth usage (completely random)

- Unfair link sharing can occur:

  Even a low bandwidth TCP connection observes packet loss which prevents it from using its fair sharing of bandwidth

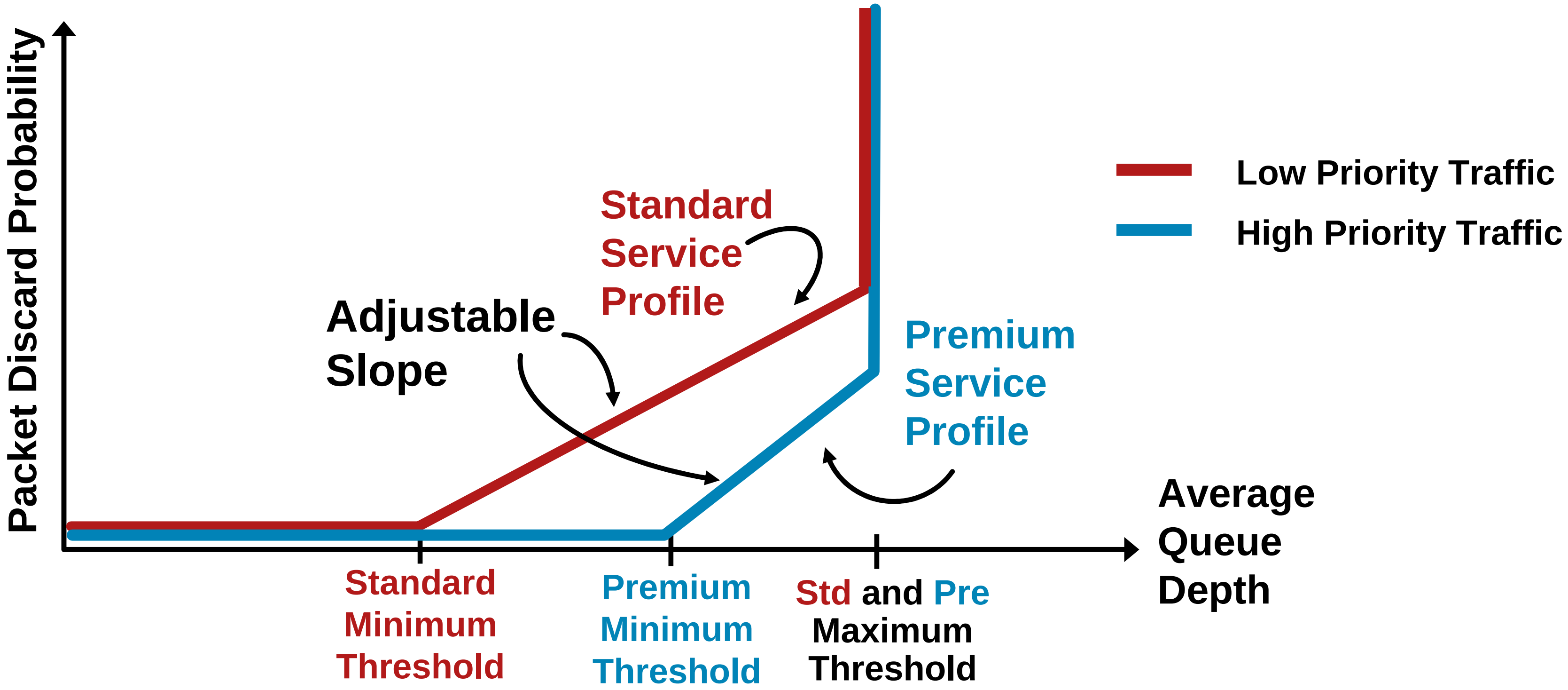
  A non-adaptive flow can increase the drop probability of all the other flows by sending at a fast rate

  The calculation of average queue length for every packet arrival is computationally intensive

Weighted Random Early Detection (WRED)

- WRED combines RED with IP Precedence to implement multiple service classes
- Each service class has a defined min and max thresholds, and drop rates
- In a congestion situation lower class traffic can be throttled back first before higher class traffic
- RED is applied to all levels of traffic to manage congestion

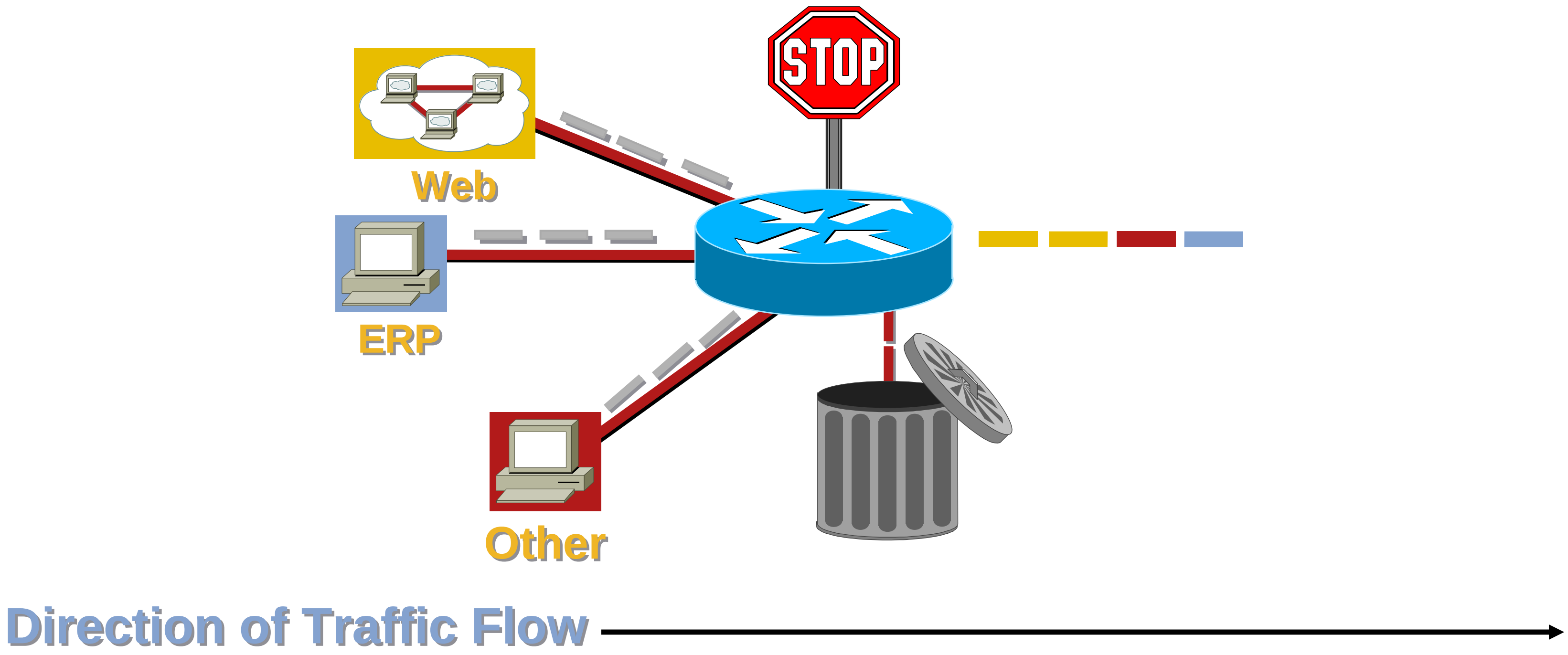
WRED Attributes for Multiple Service Levels



Two Service Levels are Shown; Up to Six Can Be Defined

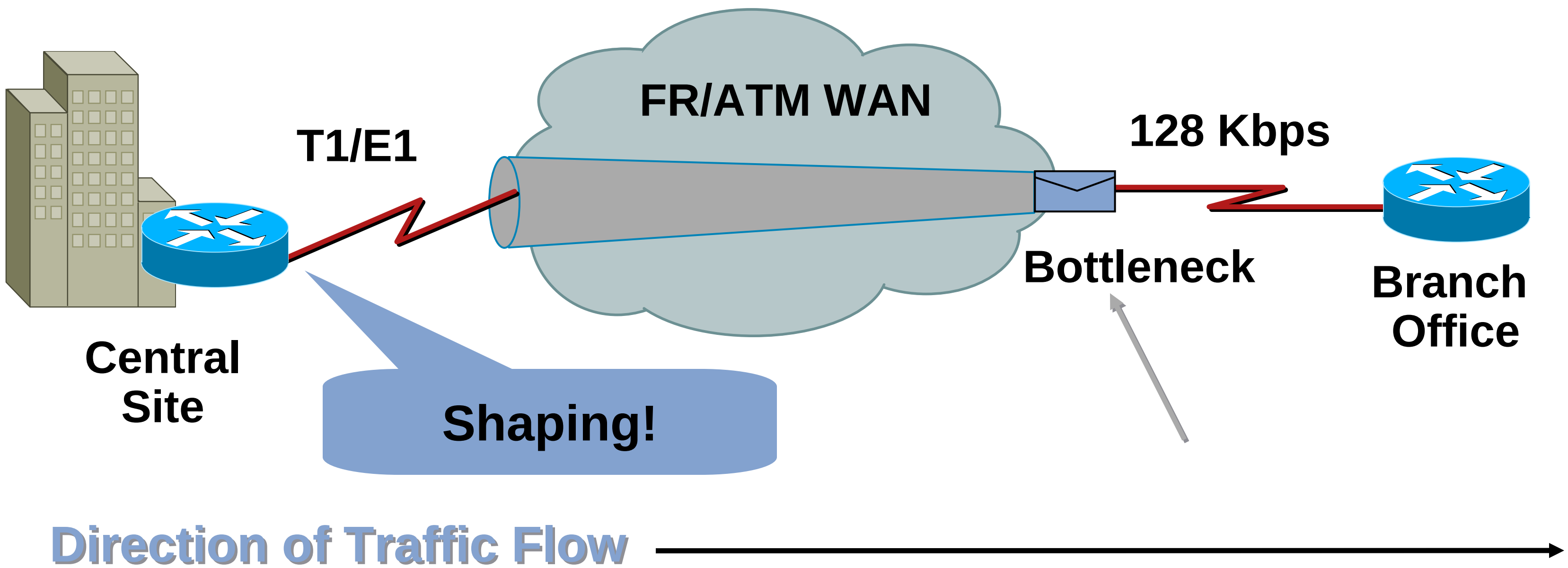
QoS Mechanisms - Policing

- **Limits traffic flow to a configured bit rate.**
- **Drops or remarks out-of-profile packets.**

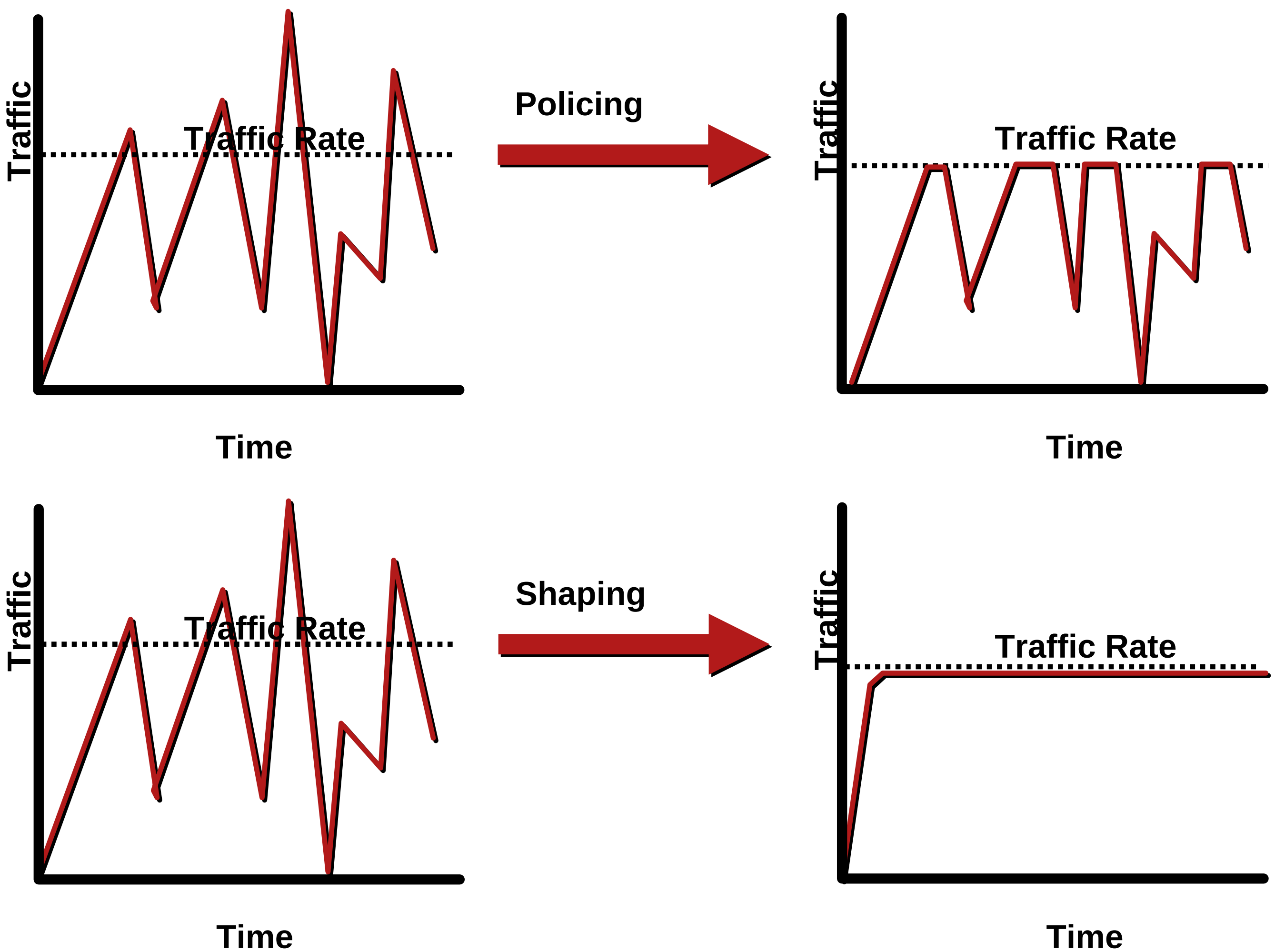


QoS Mechanisms - Shaping

- Regulates traffic flow to an average or peak bit rate.
- Commonly used where speed-mismatches exist .



Traffic Policing vs. Shaping



Traffic Policing vs. Shaping #1

	Policing	Shaping
Where Applicable	Ingress, Egress	Egress only
Buffers Excess	No	Yes
Smooths Output Rate	No	Yes
Optional Packet Remarking	Yes	No
Advantages	Controls output rate through drops. Avoids delays due to queuing.	Less likely to drop excess packets. Avoids TCP retransmissions.
Disadvantages	Drops can lead to TCP retransmits	Queuing adds delay (and jitter)

Traffic Policing vs. Shaping #2

	Policing	Shaping
Token refresh rate	Continuous based on formula: 1 / CIR	Incremented at the start of a time interval. Requires min # of intervals.
Token values	Configured in bytes.	Configured in bits per second

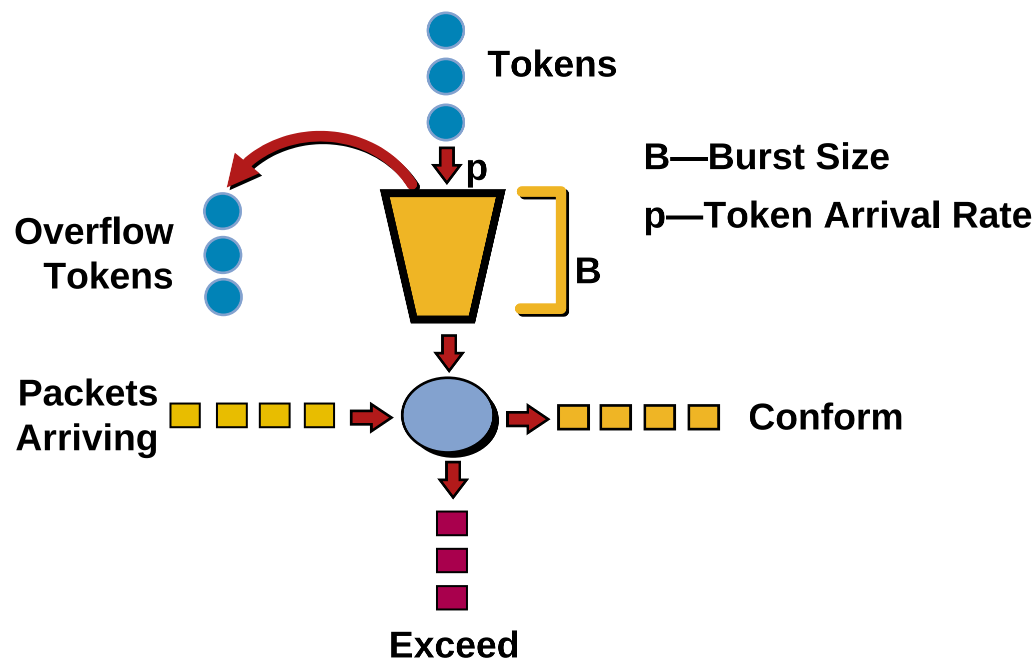
- **Both shaping and policing use the token bucket metaphor.**
- **A token bucket has no discard or priority policy.**
- **Shaping and policing differ in the rate at which tokens are replenished.**

Token Bucket Metaphor

- Tokens are put into the bucket at a certain rate.
- Each token is permission for the source to send a certain number of bits into the network.
- To send a packet, the traffic regulator must be able to remove from the bucket a number of tokens equal in representation to the packet size.
- If not enough tokens are in the bucket to send a packet, the packet either waits until the bucket has enough tokens (in the case of a shaper) or the packet is discarded or marked down (in the case of a policer).
- The bucket itself has a specified capacity. If the bucket fills to capacity, newly arriving tokens are discarded and are not available to future packets. Thus, at any time, the largest burst a source can send into the network is roughly proportional to the size of the bucket. A token bucket permits burstiness, but bounds it.

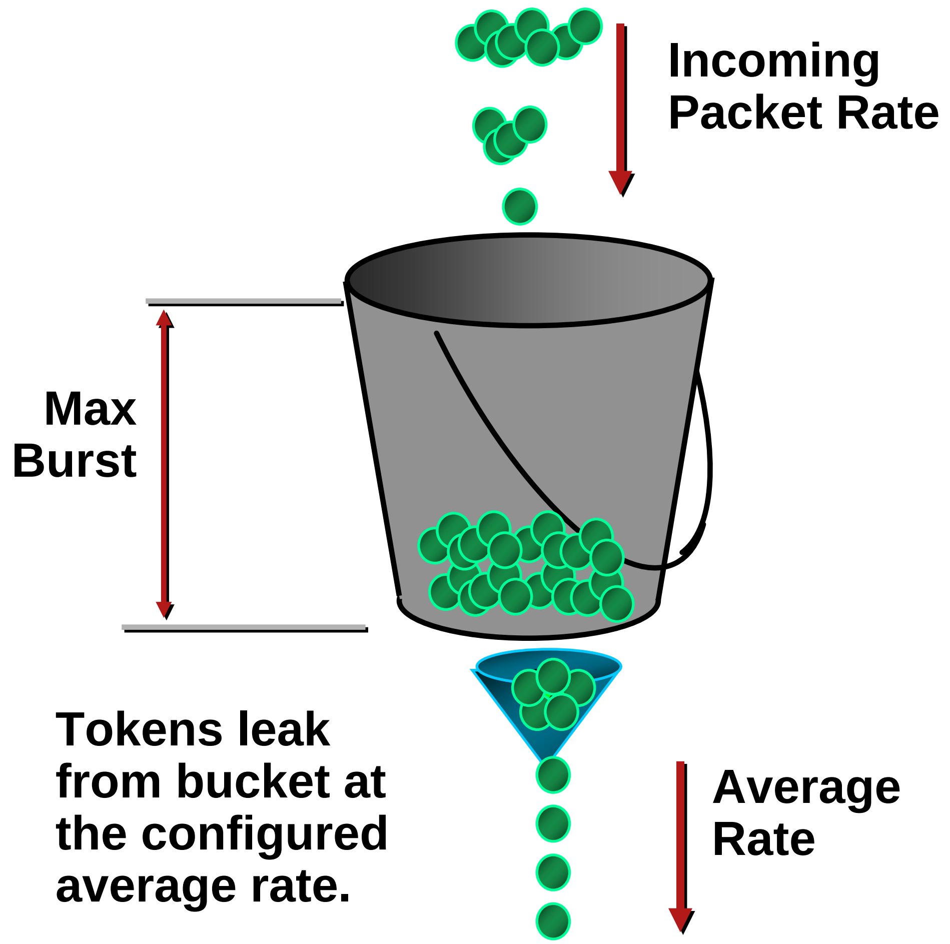
Token Bucket w/ Policing

- Tokens keep pouring into the bucket at a pre-defined average-rate
- If Token available, can transmit a packet

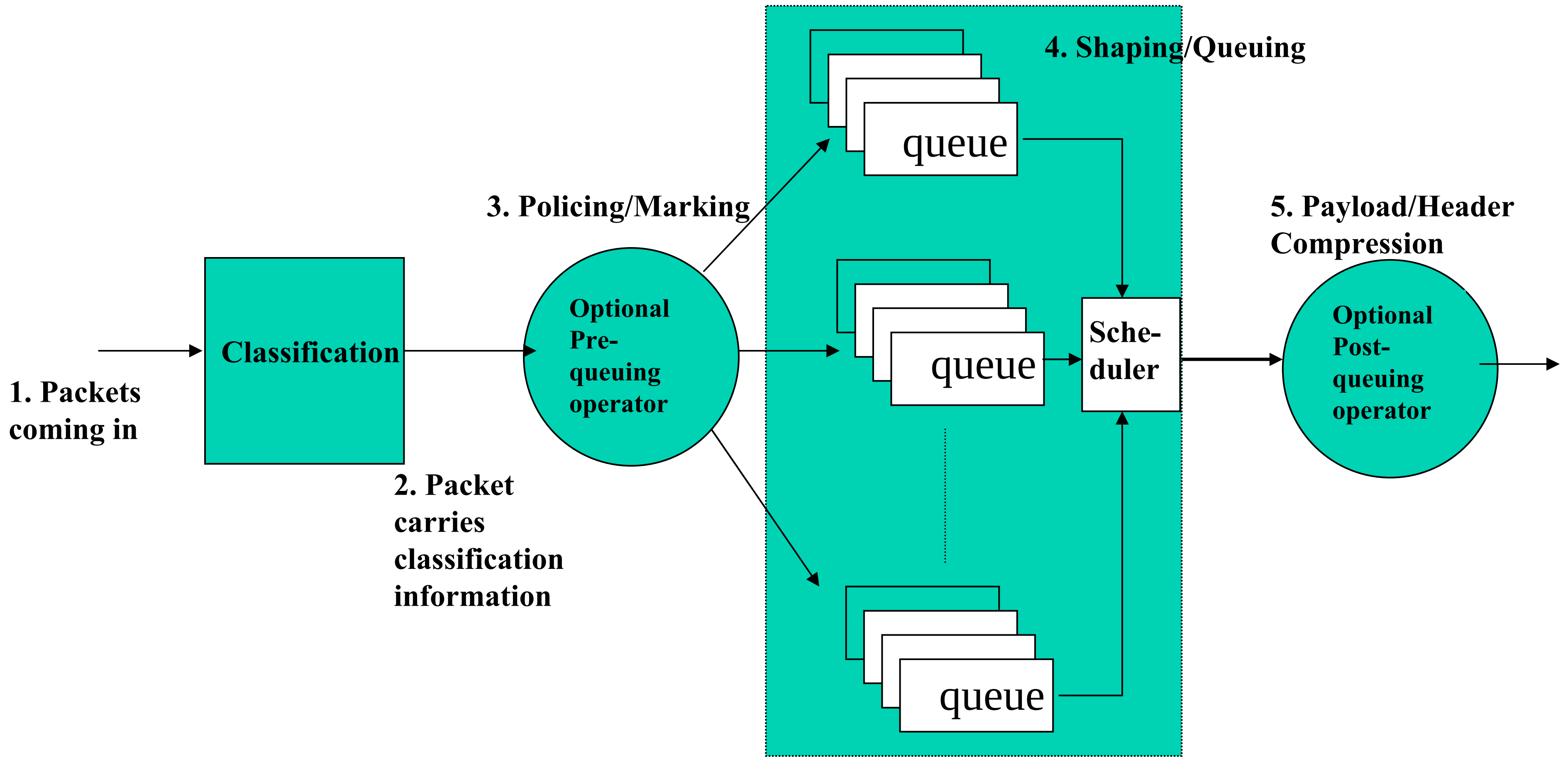


Leaky Bucket With Shaping

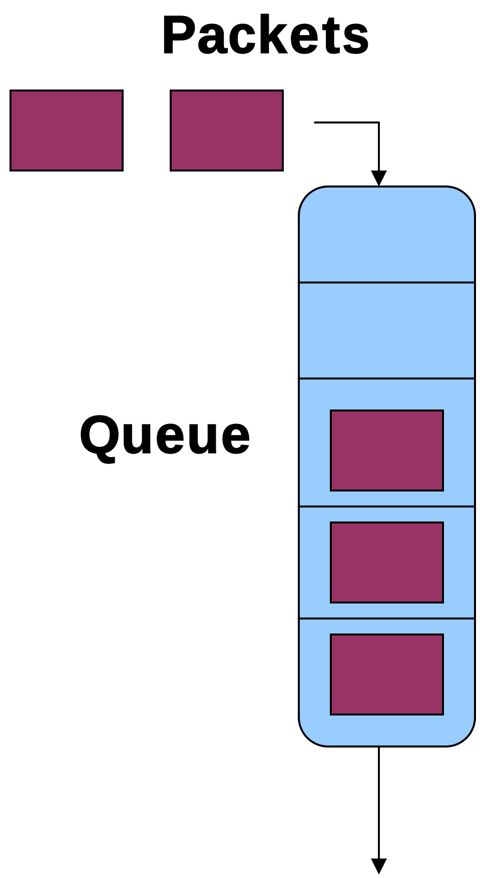
- Start with a bucket without tokens.
- Tokens can be added at a bursty rate.
- Tokens are leaked at a specified constant rate.



Putting It All Together - Packet Path



Putting it All together – Queue Definition



What controls the depth of the queue:

- Active Queue management (e.g., WRED)
- Tail drop (queue-limit)

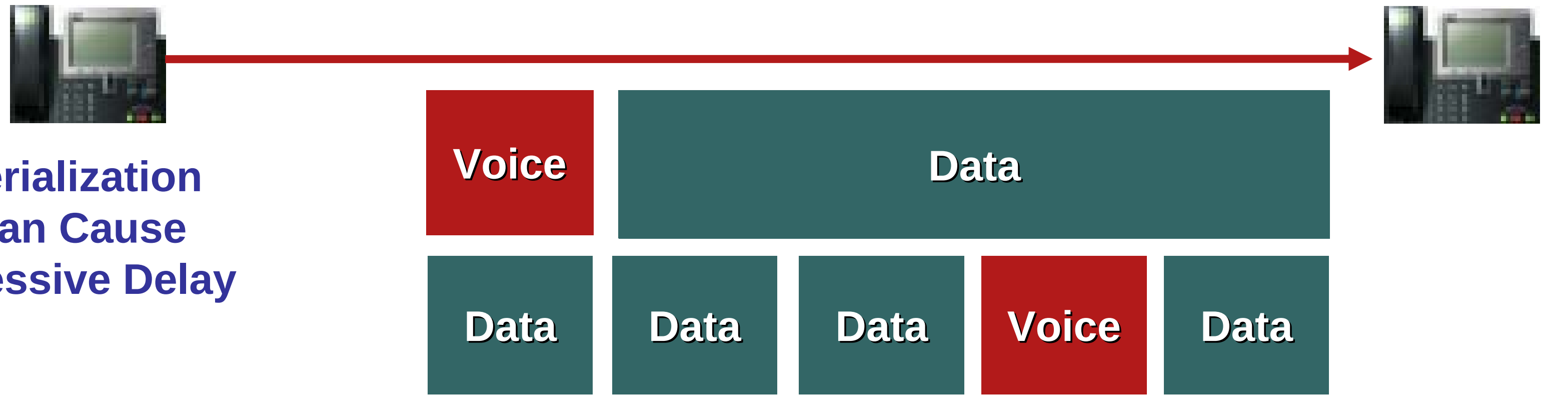
What controls the output from the queue

- Min BW guarantee
- Max BW (Shape rate)
- Excess BW (Bandwidth Remaining percent/ratio)
- Priority Level

Output from the Queue

- Priority **– low delay, strict priority queue. Gets to send its data ahead of all others queues with lower priority**. **Strictly policed to configured rate.**
- Min BW guaranteed- **the queue is guaranteed the specified BW. Gets to send before Excess BW, but after all levels of Priority traffic.**
- Excess BW (BW remaining) – **specify how to divide available BW among queues that already sent more than the Min but less than Max.**
- Max BW (Shape value) – **Shape the traffic. This is the max BW the queue receives.**

Link Efficiency Mechanisms: Link-Fragmentation and Interleaving (LFI)



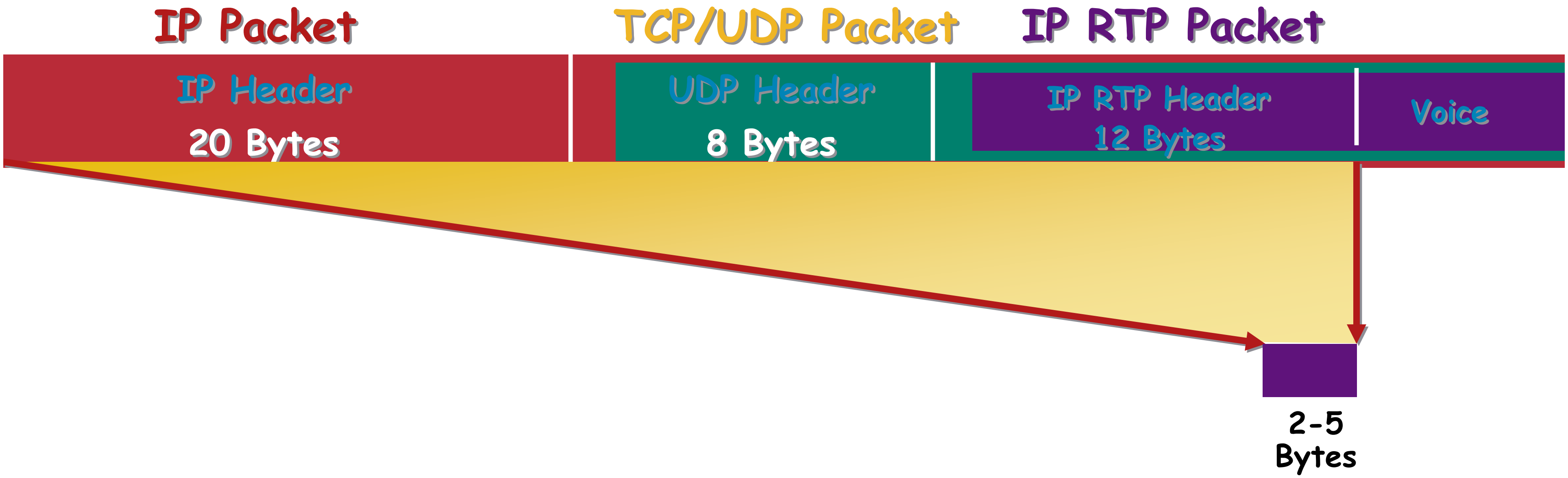
Serialization Can Cause Excessive Delay

Problem: Large Packets "Freeze Out" Voice

- Serialization delay is the finite amount of time required to put frames on a wire
- For links ≤ 768 kbps serialization delay is a major factor affecting latency and jitter
- For such slow links, large data packets need to be fragmented and interleaved with smaller, more urgent voice packets

Benefit: Reduce the Jitter and Latency in Voice Calls

Link Efficiency Mechanisms: IP RTP Header Compression



- Payload of a VoIP Packet ~ 20 bytes. But IP + UDP + RTP headers ~ 40 bytes (uncompressed)!!
- For links ≤ 768 kbps serialization delay is a major factor affecting latency and jitter
- For such slow links, large data packets need to be fragmented and interleaved with smaller, more urgent voice packets

Agenda Du Jour

- What is QoS?
- Why is it Required?
- QoS Mechanisms
- **QoS Architectures**
- Summary
- Q and A (and C)

Stateless vs. Stateful QoS Solutions

- Stateless solutions – routers maintain no fine-grained state about traffic. Example: DiffServ
  - ↑ scalable, robust
  - ↓ weak services
- Stateful solutions – routers maintain per-flow state. Example: IntServ
  - ↑ powerful services
    - guaranteed services + high resource utilization
    - fine grained differentiation
    - protection
  - ↓ much less scalable and robust

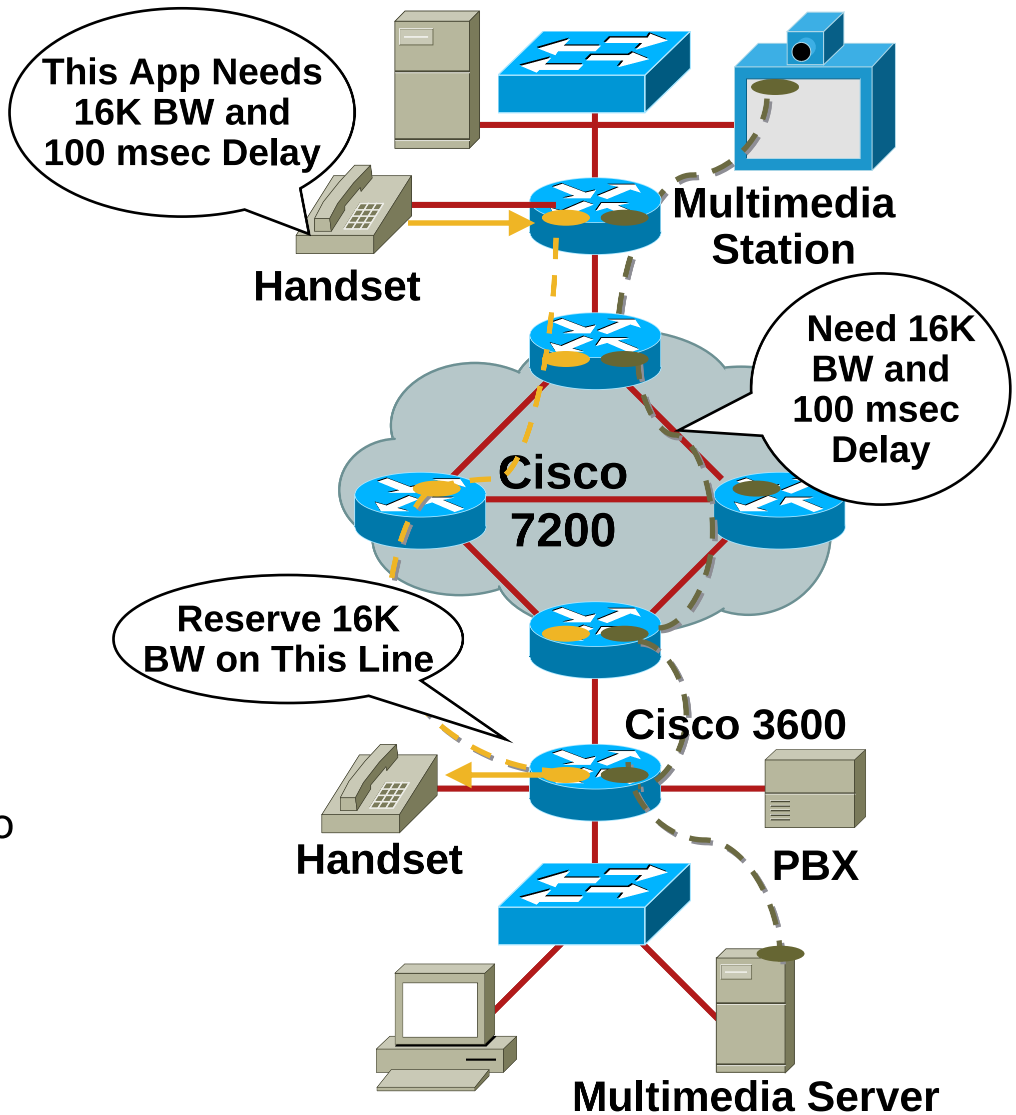
Integrated Services (IntServ) (RFCs 2210, 2211, 2212, 2215)

- An architecture for providing QOS guarantees in IP networks for individual application sessions
- Relies on resource reservation, and routers need to maintain state information of allocated resources and respond to new Call setup requests
- Key end-points are the senders and the receivers
- Applications signal their QoS requirements via a signaling protocol to the network
- Every network node along the path must check to see if the reservation request can be met
- Resources are reserved if the service constraints can be met. Reservation times out unless refreshed
- An Error message is sent back to receiver if the constraints cannot be met

Key Components of IntServ

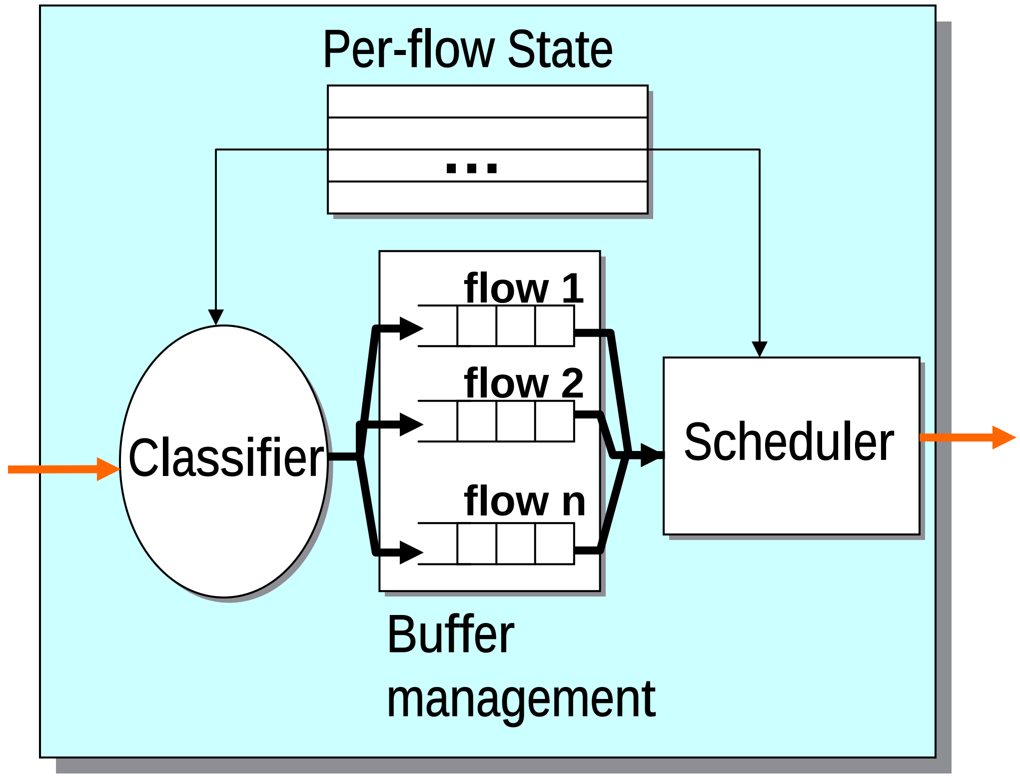
- Specification of what sender is sending: (rate, MTU, etc.)—the TSpec
- Specification of what the receiver needs: (bandwidth, path MTU, etc.)—the RSpec
- Specification of how the signalling is done to the network by the sender and the receiver

  A signaling protocol is needed to carry the R-spec and T-spec to the routers where reservation is required; RSVP is the leading candidate for such signaling protocol



Stateful Solution Complexity

- Data path
  - Per-flow classification
  - Per-flow buffer management
  - Per-flow scheduling
- Control path
  - install and maintain per-flow state for data and control paths



Stateless vs. Stateful Revisited

- Stateless solutions are more
  - scalable
  - robust
- Stateful solutions provide more powerful and flexible services
  - guaranteed services + high resource utilization
  - fine grained differentiation
  - protection

Question

- Can we achieve the best of two worlds, i.e., provide services implemented by stateful networks while maintaining advantages of stateless architectures?

    Yes, in some interesting cases. DPS, CSFQ.

- Can we provide reduced state services, I.e., maintain state only for larger granular flows rather than end-to-end flows?

    Yes: Diff-Serv

Differentiated Services (DiffServ) (RFCs 2474, 2475, 2597, 2598, 2697)

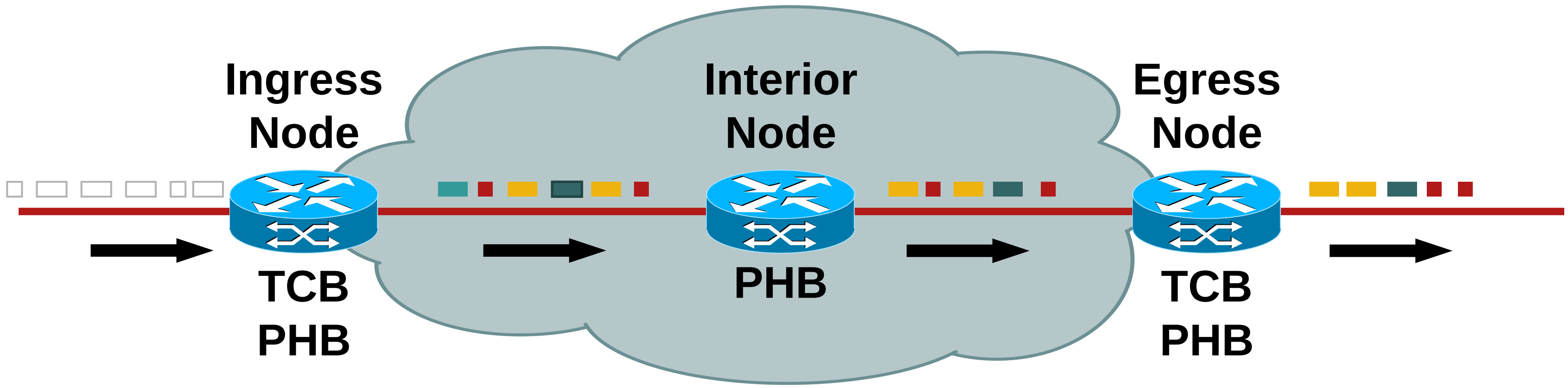
- Intended to address the following difficulties with Intserv and RSVP;

  Scalability: maintaining states by routers in high speed networks is difficult due to the very large number of flows

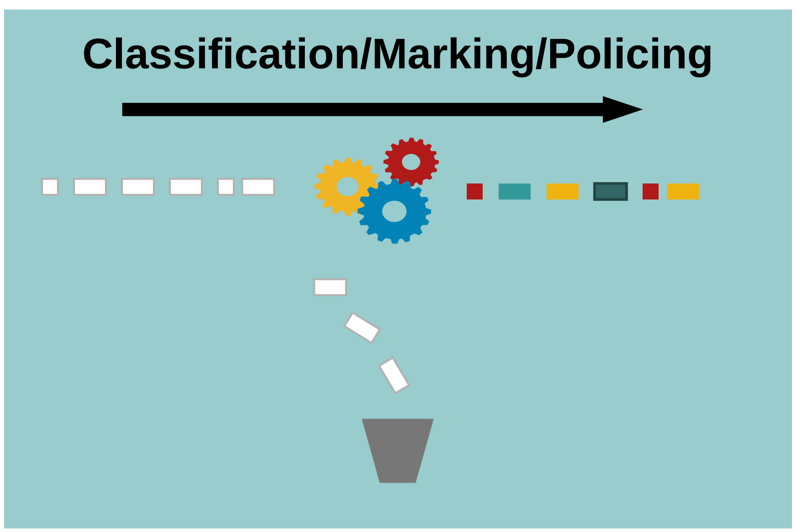
  Flexible Service Models: Intserv has only two classes, want to provide more qualitative service classes; want to provide 'relative' service distinction (Platinum, Gold, Silver, …)

  Simpler signaling: (than RSVP) many applications and users may only want to specify a more qualitative notion of service

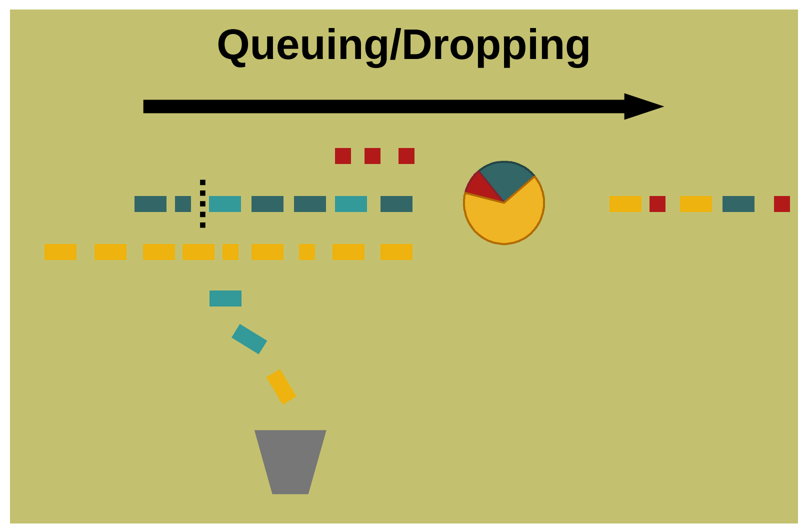
Differentiated Services Architecture (RFC 2274, RFC 2275)



Traffic Classification and Conditioning (TCB)

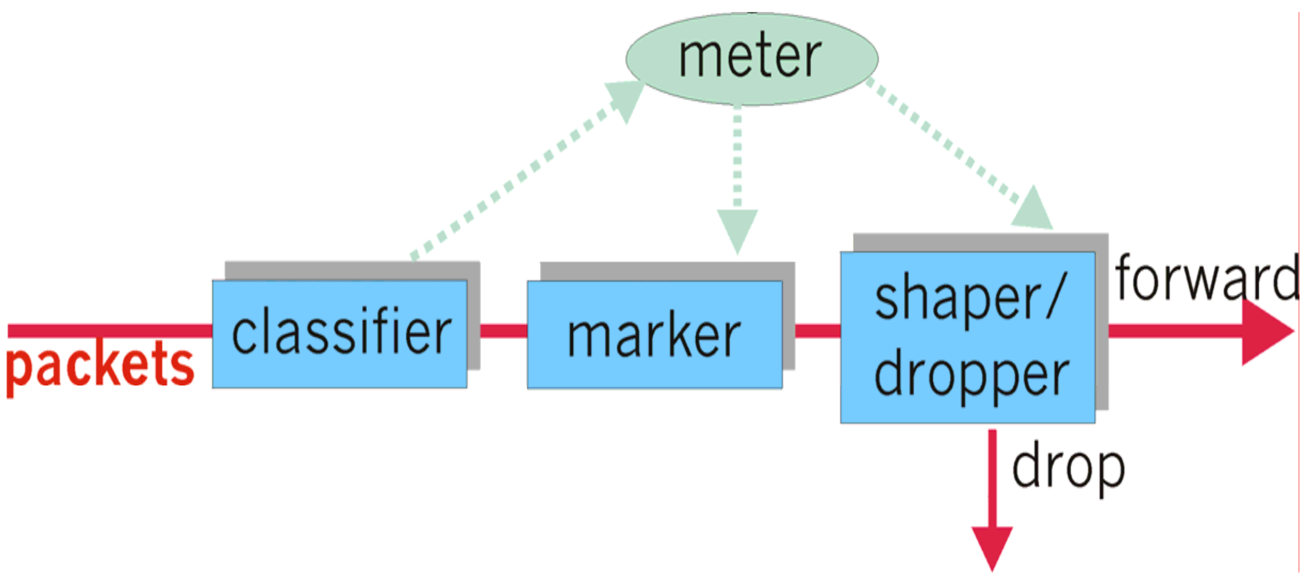


Per-Hop Behavior (PHB)



Traffic Conditioning

- It may be desirable to limit traffic injection rate of some class; user declares traffic profile (example, rate and burst size); traffic is metered and shaped if non-conforming



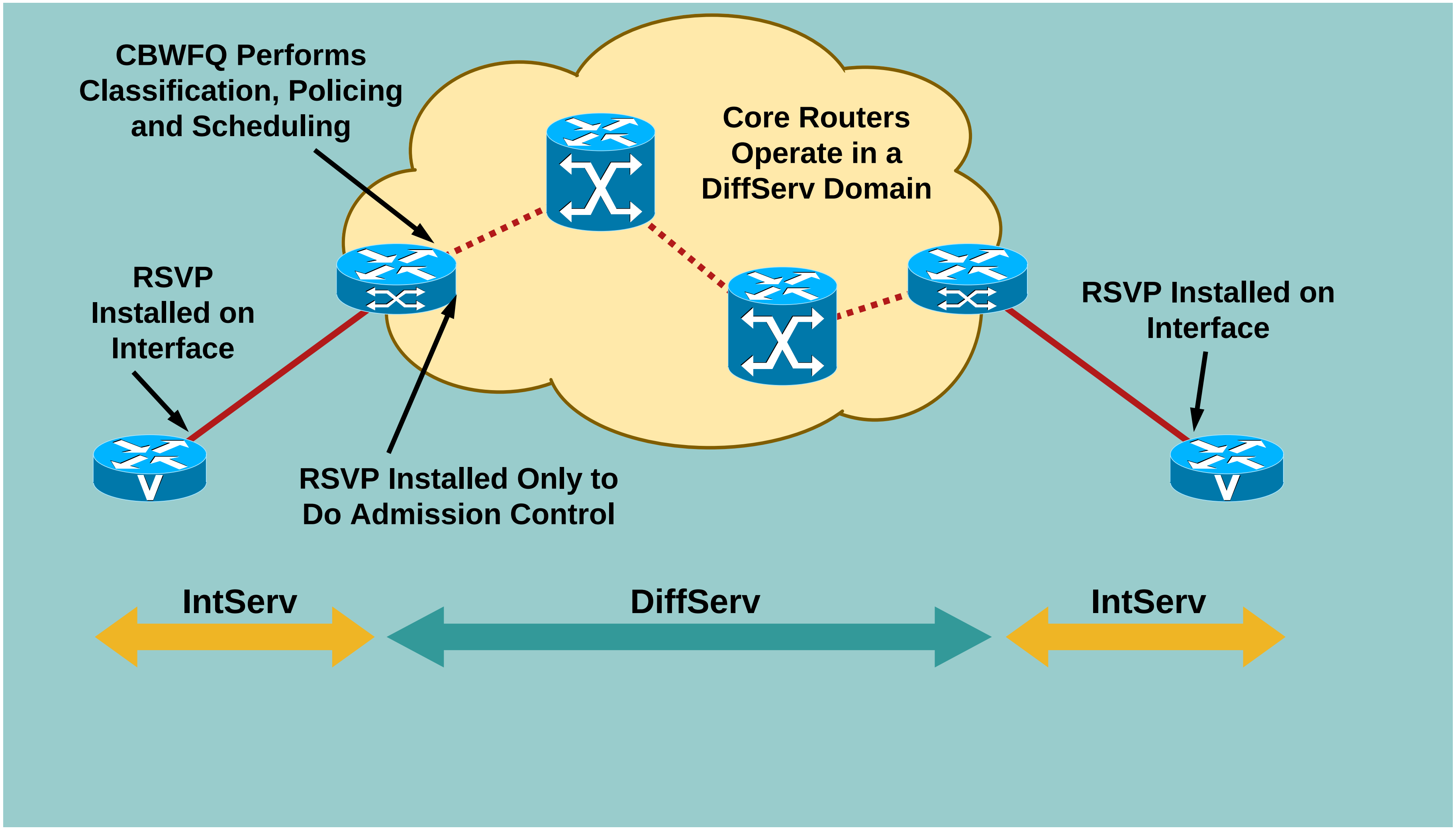
Per-hop Behavior (PHB)

- is the name for interior router data-plane functions
  - Includes scheduling, buff. mgmt, shaping etc
- Logical spec: PHB does not specify mechanisms to use to ensure performance behavior
  - Different boxes implement PHBs in different ways which are optimized for each platform
  - As long as it complies with “black box” spec, this is perfectly fine
- 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

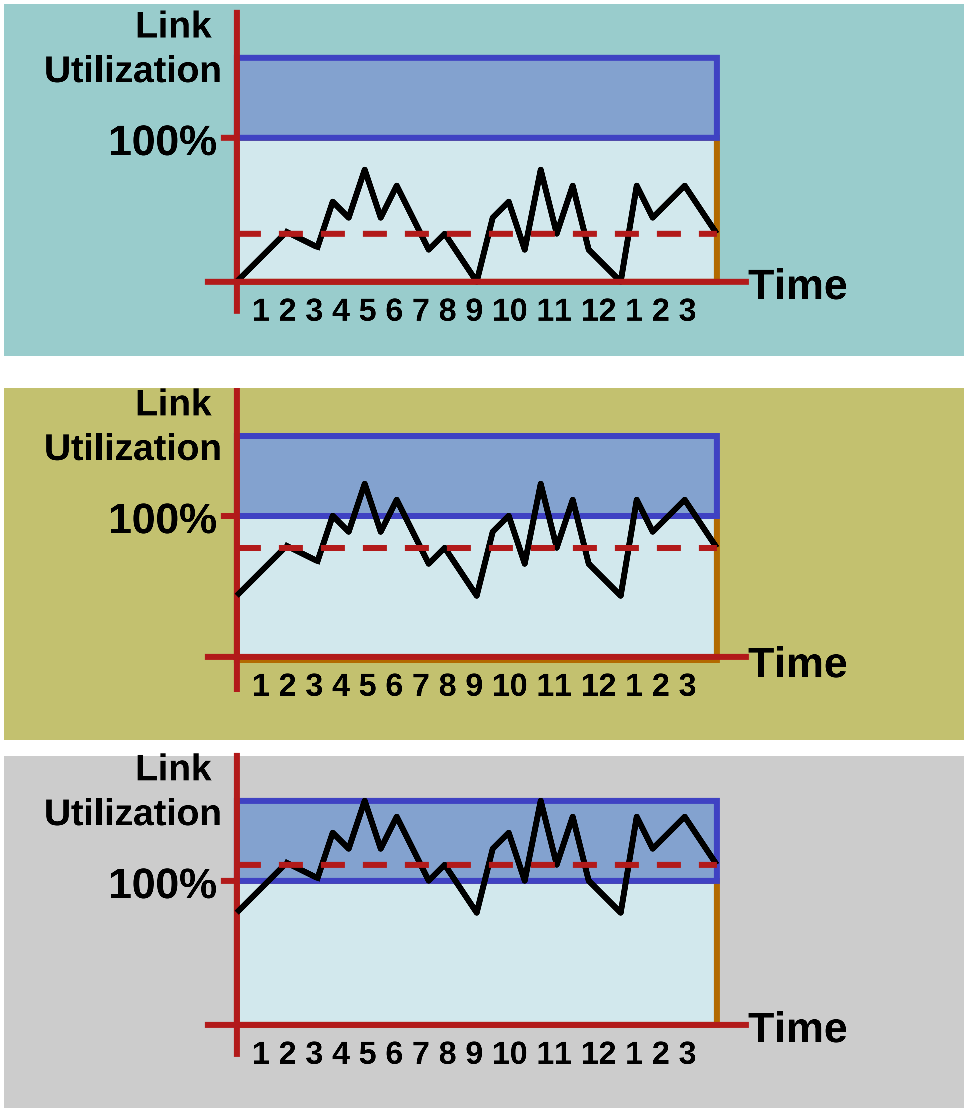
Per-Hop Behavior (contd.)

- Expedited Forwarding (EF)
  - Building block for low delay/jitter/loss
  - Served at a certain rate with short/empty queues
- Assured Forwarding (AF)
  - High probability of delivery if profile is not exceeded
  - Four classes and three levels of drop precedence
  - Specific resources (BW, buffer space) allocated to each class at each node
- Best Effort (BE)

IntServ/DiffServ Integration



QUIZ TIME !!!!!



Which of the scenarios to the left would benefit most from implementing QoS? Explain

Agenda Du Jour

- What is QoS?
- Why is it Required?
- QoS Mechanisms
- QoS Architectures
- **QoS Deployment Guide**
- Q and A (and C)

Five Steps to a Successful QoS Deployment

- Step 1: Identify and Classify Applications
  - Mission-critical apps
  - Application properties and quality requirements
- Step 2: Define QoS Policies
  - Network topology, bottleneck/non-bottleneck links
  - Trusted and untrusted boundary settings
- Step 3: Test QoS Policies
  - Baseline and Benchmarking
- Step 4: Implement Policies
  - Classify and mark close to the edge
  - Work towards the core in a phased manner
- Step 5: Monitor and Adjust

Modular QoS CLI

- MQC provides a separation between classification and features
- Platform independent way to configure QoS on cisco platforms.
- Helps in defining a QoS behavioral model. For e.g.
  - Imposing maximum transmission rate for a class of traffic
  - Guaranteeing minimum rate for a class of traffic
  - Giving low latency to a class of traffic

Hierarchical Policies

- Support for further granularity. For e.g., police aggregate tcp traffic to 10Mb/s but simultaneously police aggregate ftp traffic to 1Mb/s and http traffic to 3Mb/s

```
class-map tcp-police
    match protocol tcp
class-map ftp
    match protocol ftp
policy-map ftp-police
    class ftp
      police <bps> …
policy-map hierarchical-police
    class tcp-police
     police <bps> …
     service-policy ftp-police
```

Configuration example

class-map match-all/match-any <name>

match <filter>

policy-map <name>

class <class-name>

<feature>

Interface <interface-name>

service-policy input/output <policy-name>

As an example:

Class-map match-all precedence2

match ip precedence 2

Policy-map policy-1

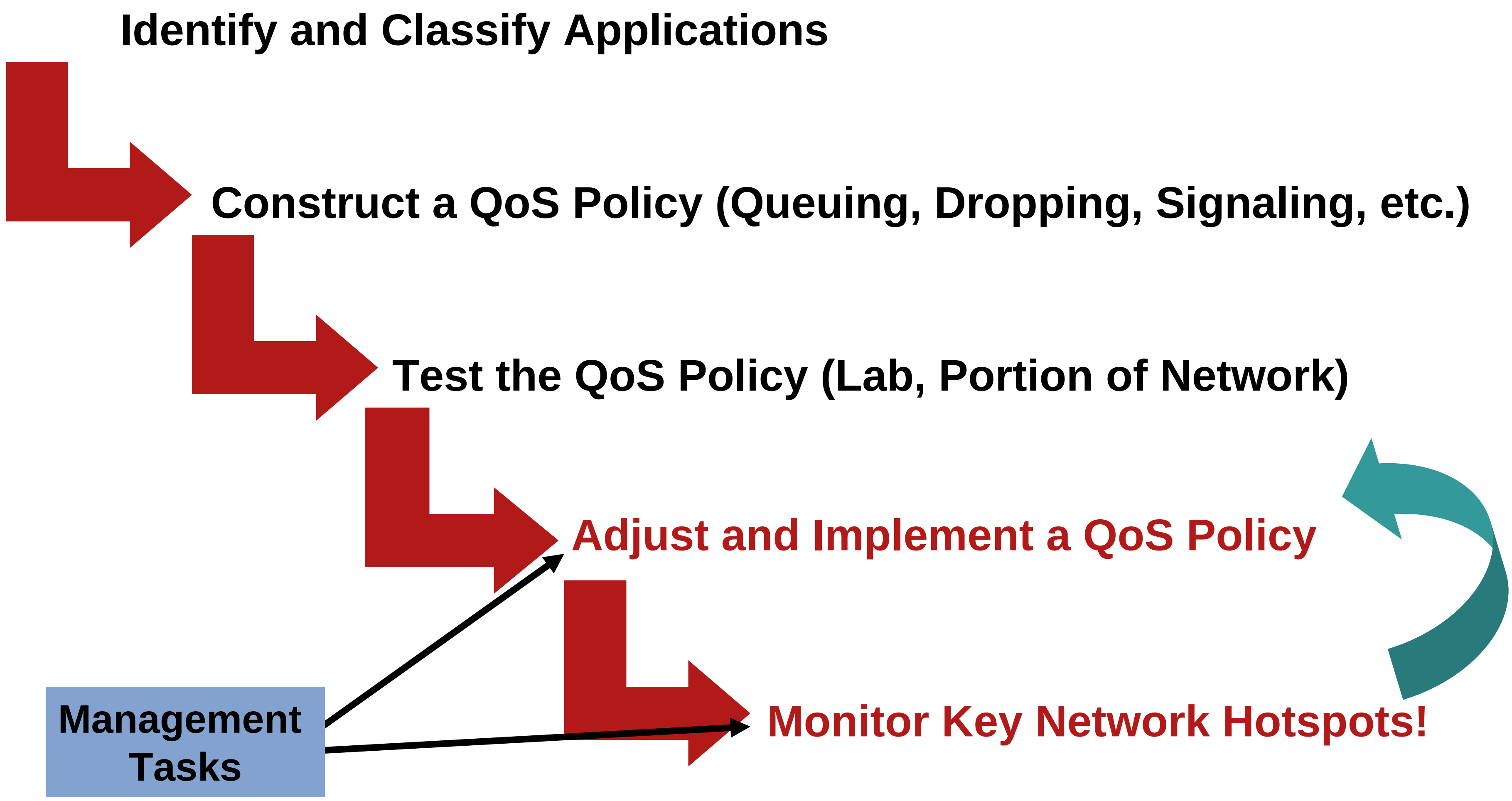
class precedence2

set ip precedence 4

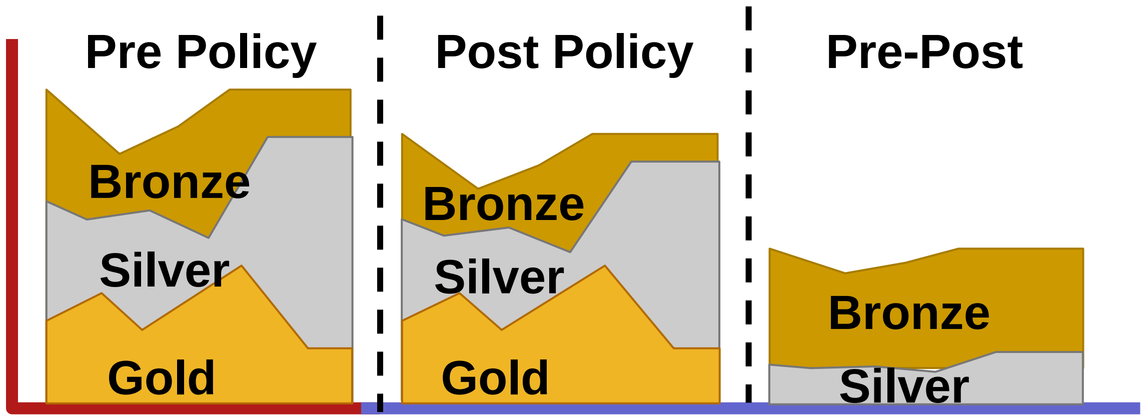
Interface Ethernet0/1

service-policy output policy-1

Remember the Five ?



Class-Based QoS MIB (CBQoSMIB)



- Primary accounting mechanism for MQC-based QoS
- Statistics for active MQC configurations on a per-policy/per-class, per-interface or PVC basis
- Monitor pre-and post-policy bit rates

  For example, "How many packets are being dropped or marked?"
- Read access only, no SNMP configuration

Cisco NBAR Protocol Discovery MIB

- Read/Write SNMP MIB support
- Real-time statistics on applications
- Per-interface, per-application, bi-directional (input and output) statistics
    - Bit rate (bps), Packet counts and Byte counts
- Top-N application views
- Application threshold settings

Cisco NBAR Protocol Discovery Statistics

```
router# sh run int fa6/0
!
interface FastEthernet0/0
 ip address 10.0.147.3 255.255.255.0
 ip nbar protocol-discovery
end

Router# show ip nbar protocol-discovery interface FastEthernet 6/0
   FastEthernet6/0
                           Input                              Output
   Protocol                         Packet Count                       Packet Count
                                    Byte Count                         Byte Count
                                    5 minute bit rate (bps)            5 minute bit rate (bps)
                                                                                         ---------
   http                             316773                             0
                                    26340105                           0
                                    3000                               0
   pop3                             4437                               7367
                                    2301891                            339213
                                    3000                               0
   snmp                             279538                             14644
                                    319106191                          673624
                                    0                                  0
   ftp                              8979                               7714
                                    906550                             694260
                                    0                                  0
   …
   Total                            17203819                           151684936
                                    19161397327                        50967034611
                                    4179000                            6620000
```


Cisco NBAR Protocol Discovery Thresholds and Traps

- User can set thresholds on individual protocols on an interface, or on a statistic regardless of protocol
  - Multiple thresholds for any combination of supported protocols/and or all protocols
- Configurable statistic types
  - Interface in, out and sum of bytes, packets, and bit rate
- If the threshold is breached, the information is stored for prolonged period of time
- A notification (trap) is generated and sent to the user with a summary of threshold information

CASE STUDY

Presentation_ID © 2006 Cisco Systems, Inc. All rights reserved. Cisco Confidential

Enterprise Network with IP Services: The WAN

- SP sells L3 services with following four levels of service
  - Real-Time
  - Business High
  - Business Low
  - Best Effort
- Business driver for Enterprise—ad-hoc any to any video conferencing from more than 60 sites across the US
  - Each site connected via T1 connection at minimum
  - VC units run standard 384Kbps IPVC streams
- Customer also has several mission critical business applications that need prioritization
- Managed CE environment

Enterprise Network with IP Services: Challenges

- Point-to-cloud model—SP is involved in QoS
- Challenges
  - Current provisioning mechanism guaranteed more than 150% of available bandwidth
  - No accounting for routing protocols and L2 overhead
  - SP not preserving DSCP marking across their cloud—Remark DSCP to indicate to themselves whether packets are within or violating contract
  - DLSW+ application configured to set its ToS value to 5 by default (same as IPVC)

Enterprise Network with IP Services: the Solution

- Customer purchased services in the ratio 5:6:2:1
- Customer migrated to a complete DSCP model
  - Simpler from a classification and provisioning perspective
  - Monitoring and management advantages
- Workaround for SP remarking: NBAR deployed at WAN edge to re-classify and re-mark INBOUND traffic from the WAN
- Routing and control traffic in business high class
- Percentage based provisioning mechanism
- QoS Policy Manager (QPM) for monitoring traffic statistics via CBQoSMIB

Enterprise Network with IP Services: Configuration

```
class-map match-all VIDEO
  match access-group 120

class-map match-all SAP
  match protocol custom-10

class-map match-all SNA
  match protocol dlsw

class-map match-all TELNET
  match protocol telnet

class-map match-all NOTES
  match protocol notes

class-map match-any WWW
  match protocol http
  match protocol secure-http

class-map match-all FTP-GRAPHICS
  match access-group 105
  match protocol ftp
```

```
class-map match-all REAL-TIME
  match ip dscp ef

class-map match-any BUSINESS-HIGH
  match ip dscp af31
  match ip dscp af32
  match ip dscp af33
  match ip dscp cs3

class-map match-any BUSINESS-LOW
  match ip dscp af21
  match ip dscp af22
  match ip dscp af23
```

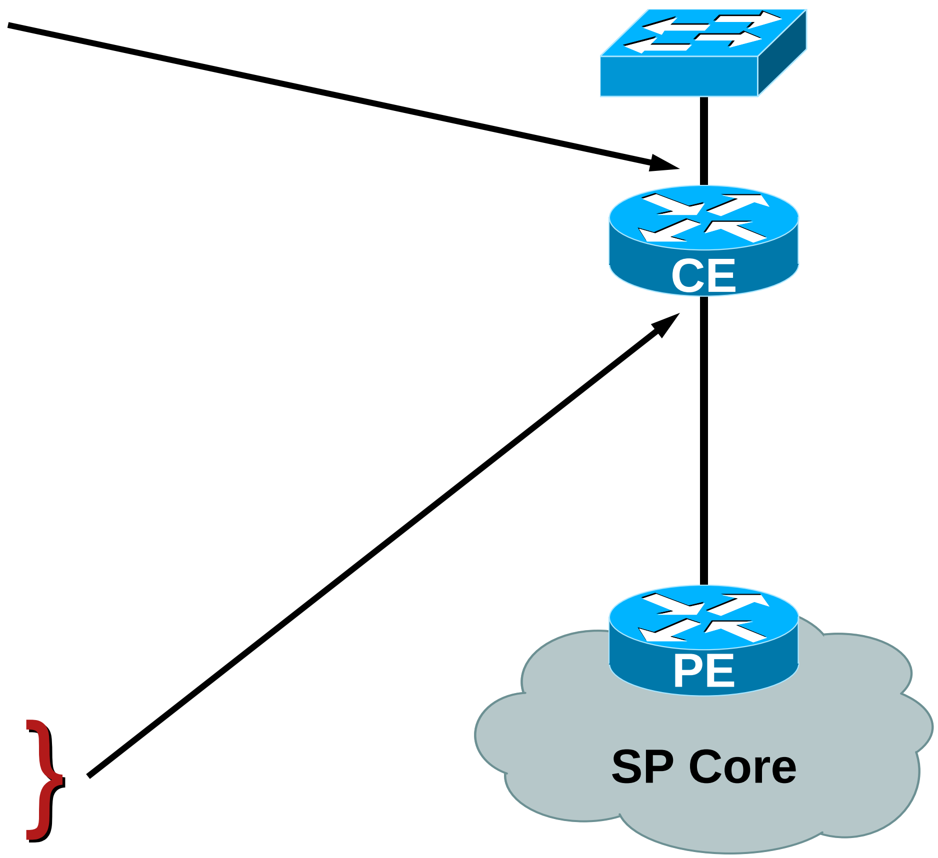

Enterprise Network with IP Services: Configuration (Cont.)

```
policy-map MARKING
 class VIDEO
  set ip dscp ef
 class SAP
  set ip dscp af31
 class SNA
  set ip dscp af32
 class TELNET
  set ip dscp af33
 class NOTES
  set ip dscp af21
 class WWW
  set ip dscp af22
 class FTP-GRAPHICS
  set ip dscp af23
 class SCAVENGER
  set ip dscp cs1
 class class-default
  set ip dscp default
```

```
policy-map WAN-EDGE
 class REAL-TIME
  priority 512
 class BUSINESS-HIGH
  bandwidth percent 45
  random-detect dscp-based
 class BUSINESS-LOW
  bandwidth percent 15
  random-detect dscp-based
 class SCAVENGER
  bandwidth percent 1
 class class-default
  fair-queue
  random-detect dscp-based
```


Enterprise Network with IP Services: Configuration (Cont.)

```
interface FastEthernet0/0
 service-policy input MARKING
!
interface Serial0/0
 encapsulation frame-relay IETF
 frame-relay traffic-shaping
!
interface Serial0/0.1 point-to-point
 description SP Ckt
 frame-relay interface-dlci 101
  class FRTS
!
map-class frame-relay FRTS
 frame-relay cir 1536000
 frame-relay bc 15360
 frame-relay mincir 1536000
 service-policy input MARKING
 service-policy output WAN-EDGE
```



Deployment Guide: Cheat Sheet

- Aggregation and speed transition links are potential choke points
- Buffer management, marking and policing in the campus, access and distribution layers
- Protect mission critical applications first
- Single class for latency sensitive traffic, additional traffic classes to implement data SLAs
- Optional class for routing and management traffic
- Less than best effort service for scavenger (P2P, worms) class
- Most other application traffic falls in Best-Effort class
- Queuing and shaping enabled at the egress WAN edge
- Remarking and policing enabled at the ingress provider edge
- Queuing and WRED dropping enabled in the SP core

How Many More Slides ???

MYTH: This presentation could go on forever!

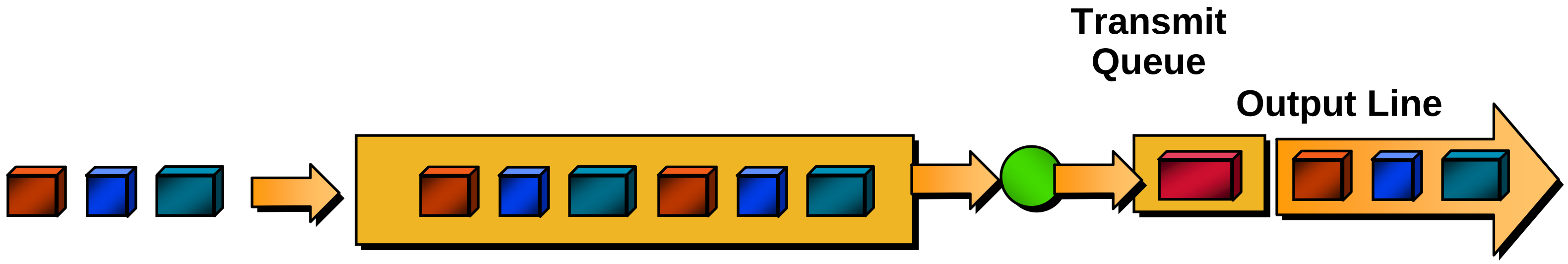
FACT: It's over, but there's a lot more to QoS…(next year, once I learn it first ☺)!!

Q and A

And (C)omments

Backup Slides

Queuing Algorithms - FIFO



- Simplest Queuing Algorithm
- “packets leave in order of arrival”
- Fixed Queue Lengths (default in IOS = 40)

  Result in dropping from tail of queue under load
- Bursty sources may cause high delay in delivering time-sensitive control/signaling messages

Queuing Algorithms – Priority Queuing (PQ)

- Assigns packets to one of four queues (high/medium/default-normal/low)
- Servicing is always top-down; Higher queues are completely exhausted before lower queues are serviced
- Excellent protection for latency sensitive traffic
- BUT
  - **FIFO drawbacks within PQ**
  - **Starvation between PQ's**
  - **Human analysis / configuration**

Queuing Algorithms – Custom Queuing (CQ)

- Reserves a portion of the bandwidth of a link for each selected traffic type
- up to 16 queues defined and traffic-share counts are assigned to each queue
- mitigates starvation scenarios by introducing the concept of “guaranteed minimum” bandwidth
- FIFO within CQ, RR between CQ’s

a) cycle through the series of queues in round-robin order

b) send the portion of allocated bandwidth for each queue before moving to the next queue

c) Queuing of packets is still FIFO in nature in each classification

- BUT
  - **FIFO drawbacks within PQ**
  - **Human analysis / configuration**

Queuing Algorithms – Weighted Fair Queuing (WFQ)

- An algorithm is a Fair Queuing Algorithm (FQ) iff:

  it sorts data streams by conversation (flow)

  data streams that use less of the interface bandwidth are algorithmically guaranteed as much bandwidth as they demand with minimal latency;

  data streams that use more are algorithmically guaranteed to use approximately the same bandwidth, with potentially increased latency.

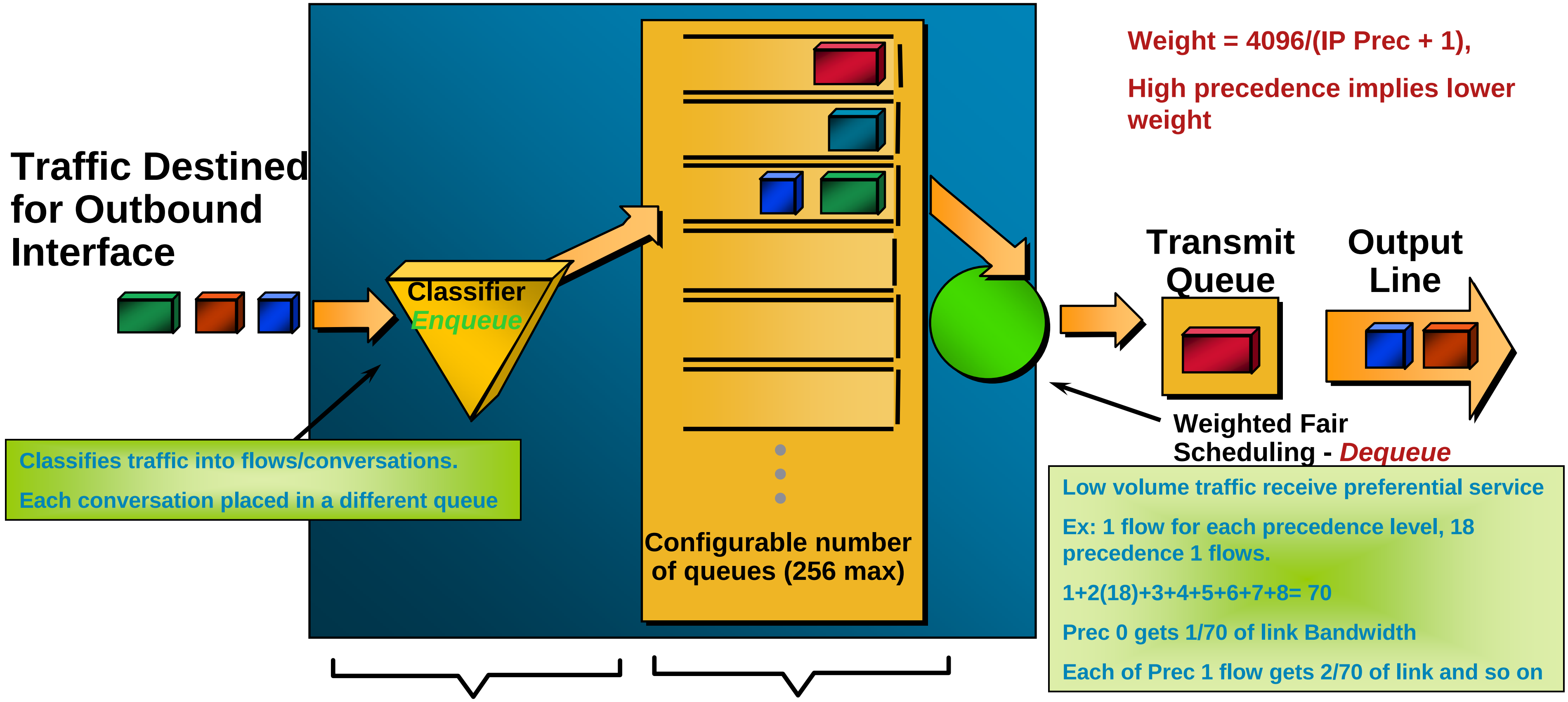
- An algorithm is a Weighted Fair Queuing Algorithm (WFQ) iff

  it is a Fair Queuing algorithm after a per-stream multiplier is applied to the bandwidths of the streams.

- WFQ is similar in some respects to Custom Queuing.

  The big difference is that it sorts among individual traffic streams without having the user define access lists.

Weighted Fair Queuing - Operation



Flow-Based Classification by:

- Source and destination address
- Protocol
- Session identifier (port/socket)

Queuing Buffer Resources

Weight Determined by:

- Requested QoS (IP Precedence, RSVP)
- Frame Relay FECN, BECN, DE (For FR Traffic)

RED – Packet Drop Probability

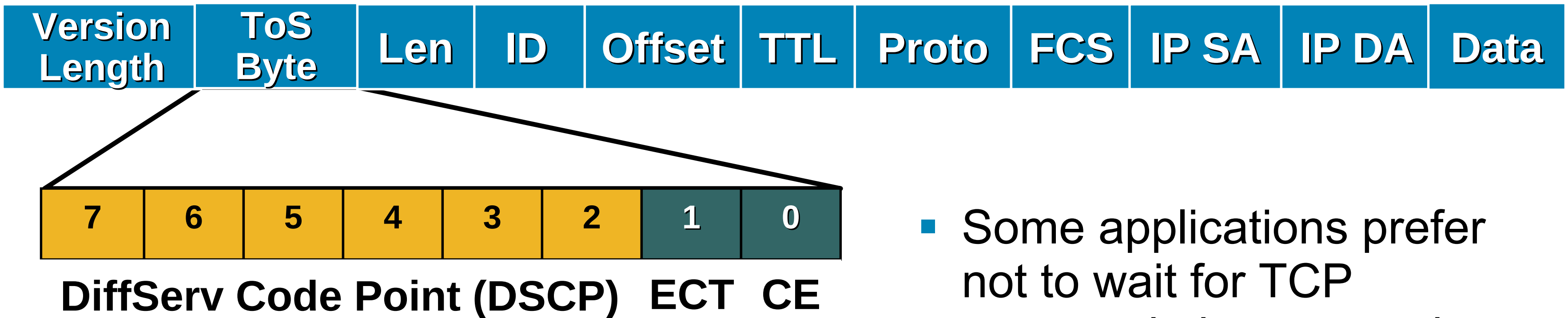
- The packet drop probability is based on the minimum threshold, maximum threshold, and mark probability denominator
- The minimum threshold value should be set high enough to maximize the link utilization. If the minimum threshold is too low, packets may be dropped unnecessarily, and the transmission link will not be fully used
- The difference between the maximum threshold and the minimum threshold should be large enough to avoid global synchronization. If the difference is too small many packets may be dropped at once, resulting in global synchronization
- WRED tuning depends upon many factors, including:

    The offered traffic load and profile

    The ratio of load to available capacity

    The behaviour of traffic in the presence of congestion

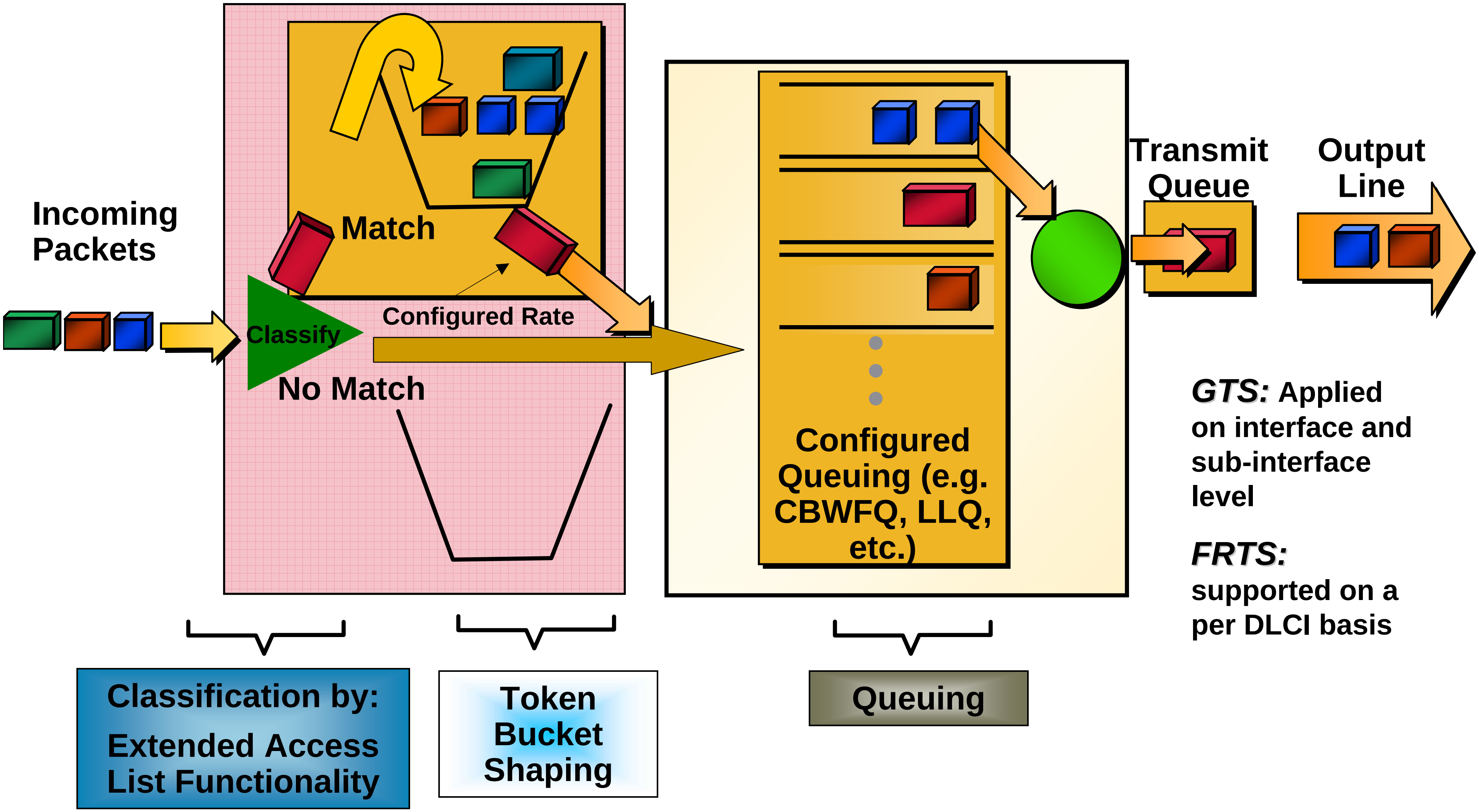
Weighted Random Early Detection: Explicit Congestion Notification (ECN)



Non ECN-Capable (ECT, CE)	0	0
ECN Capable Endpoints (ECT)	0	1
ECP Capable Endpoints (ECT)	1	0
Congestion Experienced (ECT,CE)	1	1

- Some applications prefer not to wait for TCP retransmit timer to expire
  - Short web transfers and low bandwidth Telnet
- No packet drop
  - Congestion notification signal is sent to end host

Generic Traffic Shaping (GTS)



GTS: **Applied on interface and sub-interface level**

FRTS: **supported on a per DLCI basis**

Major Types of AQM - RED (RED Variants)

- WRED (Weighted-RED)
  Profiles packet with different probabilities at the same level of congestion

- ARED (Adaptive-RED)
  Attempts to maintain suitable operating parameters in RED by dynamically adjusting maxp (max of Pb)

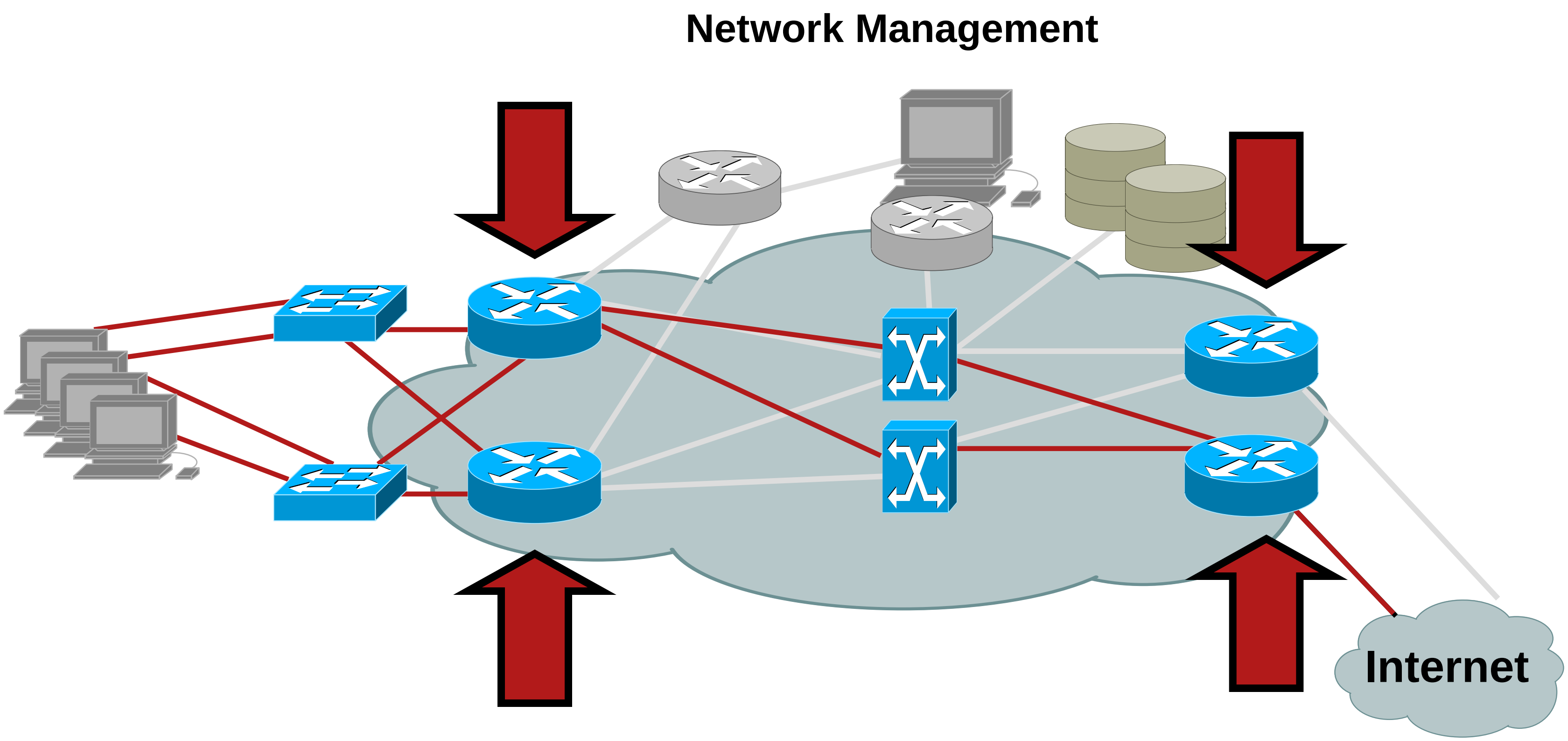
- DRED (Dynamic-RED)
  Adjusts the packet drop probability based on the deviation of the queue length

- SRED (Stabilized-RED)
  Stabilizes the buffer utilization at a level independent of the load level

Where is WRED used?

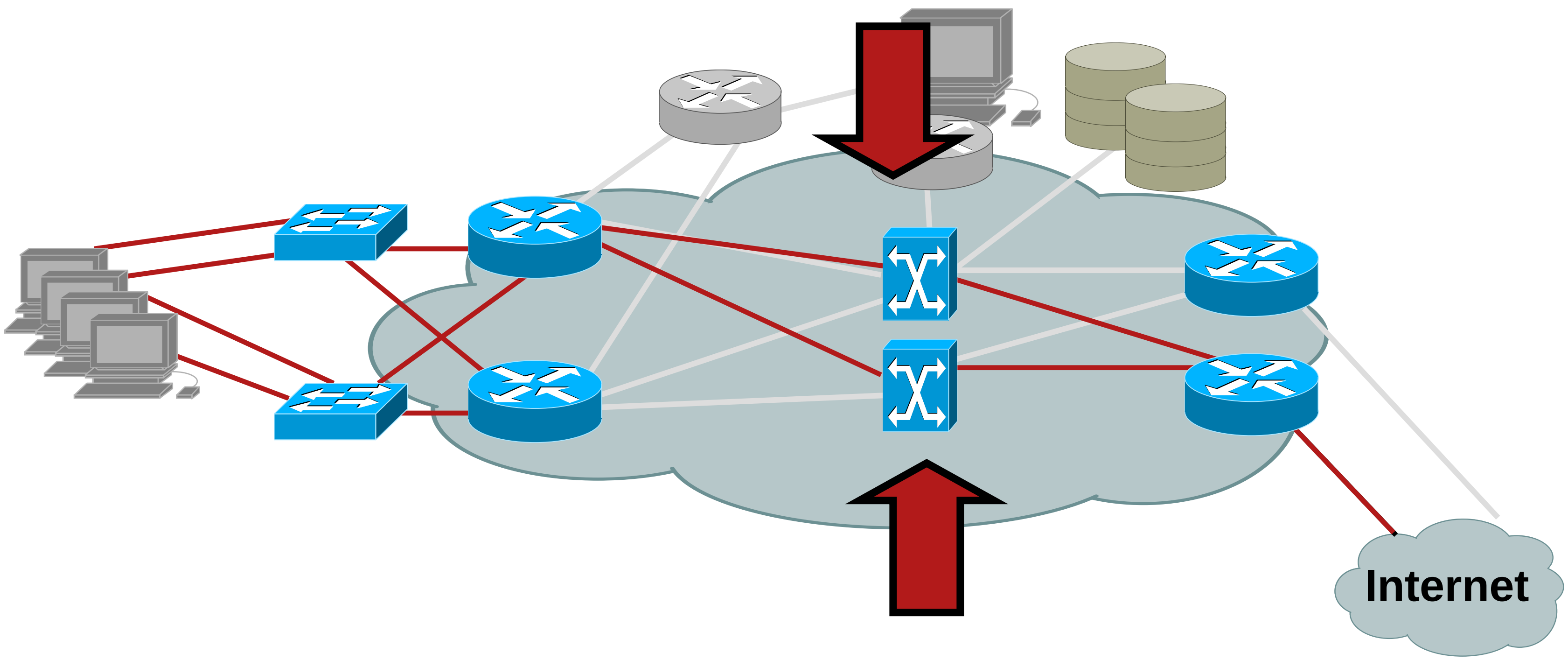
- WRED is useful on any output interface where you expect to have congestion
- WRED is usually used in the core routers of a network, rather than the network's edges
- Edge routers assign IP precedence to packets as they enter the network
- WRED uses these precedences to treat different types of traffic
- When the bulk of your traffic is TCP as opposed to UDP (Why?)

Classifying and Marking



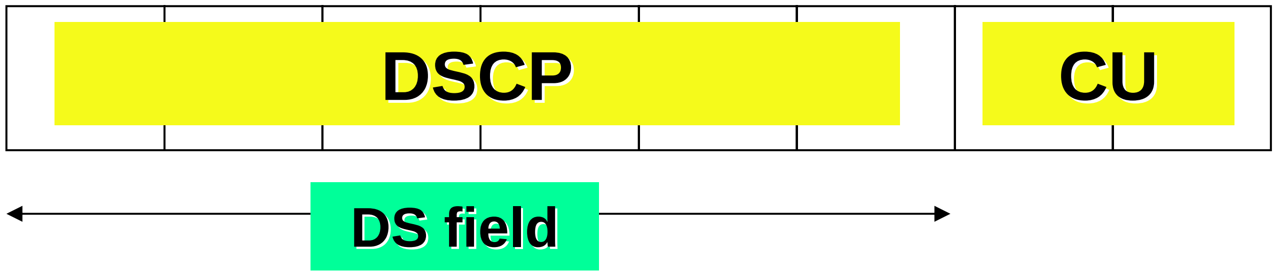
Classification and marking of packets at the edge of the network makes the packets accessible to QoS handling within the network

Optimized Forwarding



Optimized queuing and forwarding in the core of the network (PHB – Per Hop Behavior) allows for fast efficient delivery

Differentiated Services Code Point (DSCP)



- Packet is marked in the Type of Service (TOS) in IPv4, and Traffic Class in IPv6: renamed as "DS"
- DSCP : Differentiated Service Code Point = 6 bits
- CU: Currently Unused = 2 bits (lined up for ECN)
- DSCP is the field identifying what treatment (PHB) the packet should receive

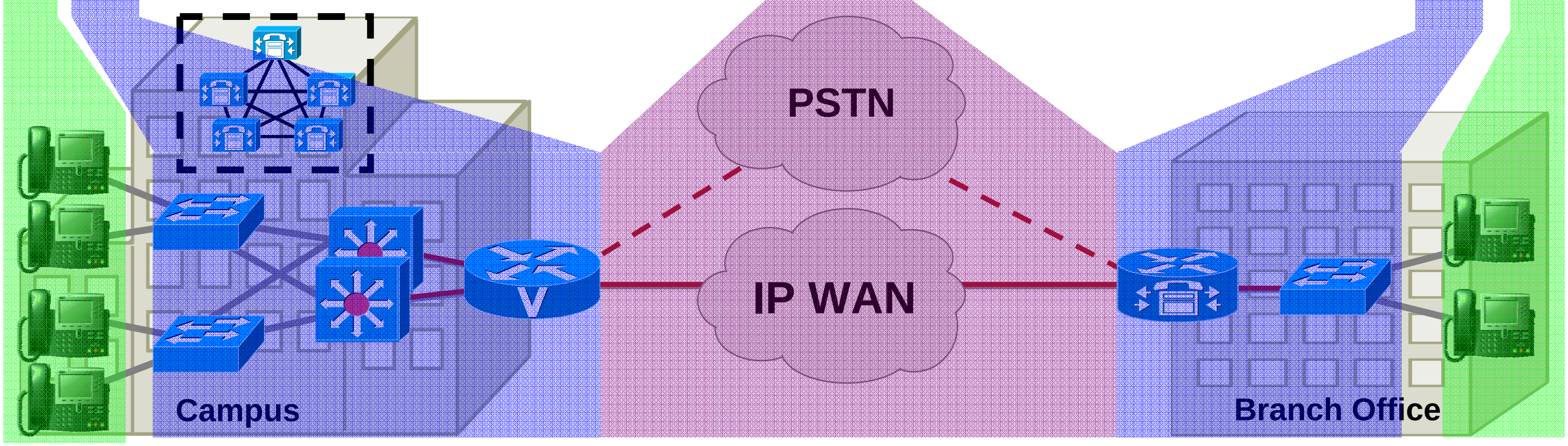
Design Approach to Enabling QoS

Classification: Mark the packets with a specific priority denoting a requirement for class of service from the network

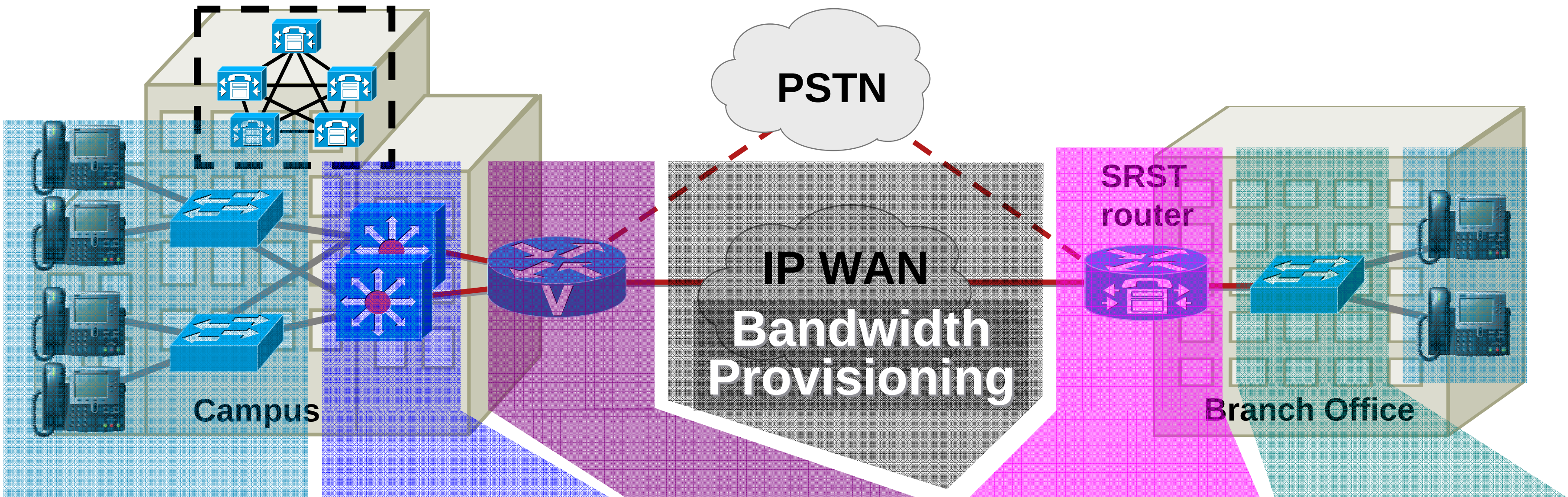
Trust Boundary: Define and enforce a trust boundary at the network edge

Scheduling: Assign packets to one of multiple queues (based on classification) for expedited treatment throughout the network; use congestion avoidance for data

Provisioning: Accurately calculate the required bandwidth for all applications plus element overhead



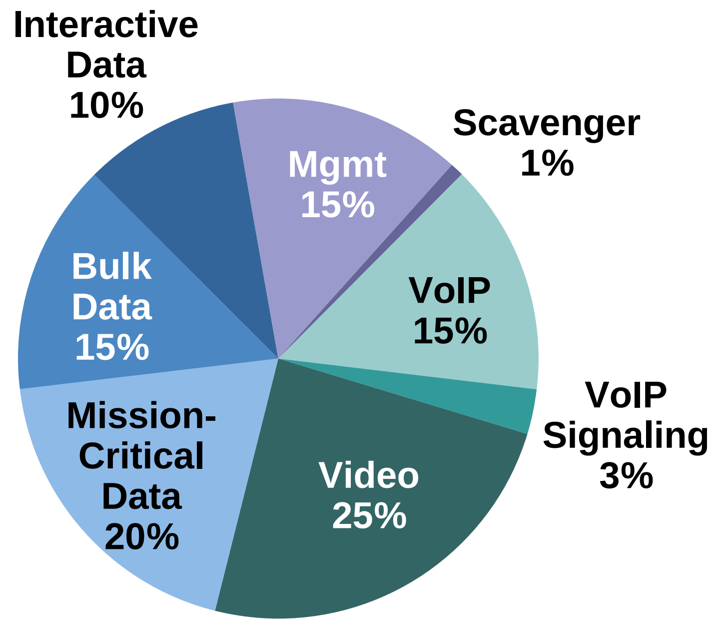
QoS Tools Mapped To Design Requirements



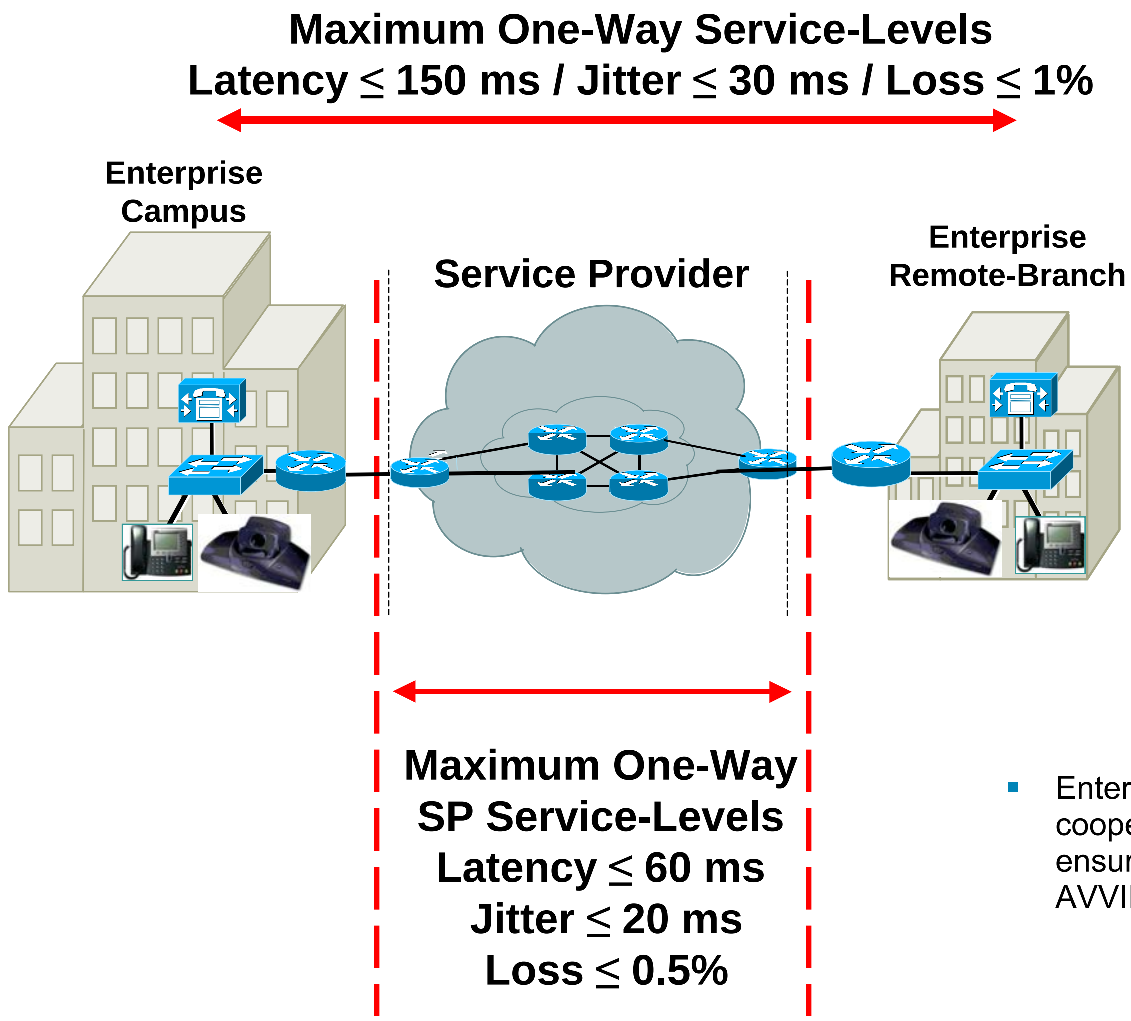
Campus Access	Campus Distribution/Core	WAN Aggregator	Branch Router	Branch Switch
• Auxiliary VLANs • Multiple Queues • 802.1Q/p • DSCP • Fast link convergence • Strict Priority Queuing	• Multiple Queues • 802.1Q/p • DSCP • Strict Priority Queuing	• DSCP • LLQ • CBWFQ • WRED • LFI/FRF.12 • cRTP • FRTS, dTS	• DSCP • NBAR • 802.1Q/p • LLQ • CBWFQ • WRED • LFI/FRF.12 • cRTP • FRTS	• Auxiliary VLANs • Multiple Queues • 802.1Q/p • Strict Priority Queuing

Queuing: Sample Policy for WAN Bandwidth Allocation

```
policy-map Multiservice
  class VoIP
    priority percent 15
  class VoIP-Signaling
    bandwidth remaining percent 3
  class video
    bandwidth remaining percent 25
  class Mission-Critical-Data
    bandwidth remaining percent 20
  class Bulk-Data
    bandwidth remaining percent 15
  class Interactive-Data
    bandwidth remaining percent 10
  class Management
    bandwidth remaining percent 15
  class Scavenger
    bandwidth remaining percent 1
  class class-default
    fair-queue
```



Service-Provider Considerations



- Enterprises and SPs must cooperate and be consistent to ensure QoS requirements for AVVID