Introduction to QOS

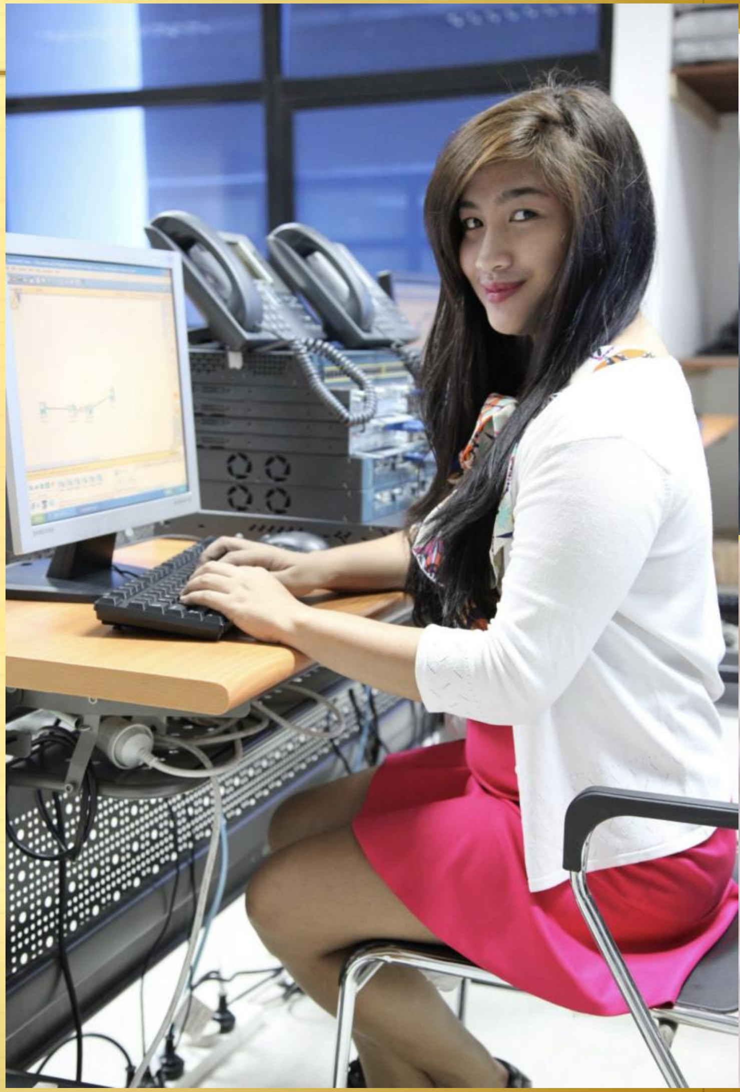
Agenda

- Introduction to QOS
  - What is QOS?
  - QOS models
  - QOS operations
  - QOS design principles
- QOS for convergence
  - Voice, video, data QOS requirements
  - QOS technology review (classification, policing and scheduling tools)
- IOS QOS implementation
  - MQC
  - AutoQos
- QOS for security

QOS introduction

What Is Quality of Service?

- To the end user
  - User's perception that their applications are performing properly
  - Voice – No drop calls, no static
  - Video – High quality, smooth video
  - Data – Rapid response time
- To The Network Manager
  - Need to maximize network bandwidth utilization while meeting performance expectations of the end user
  - Control Delay, Jitter, and Packet Loss



Different Types of Traffic Have Different Needs

- Real-time applications especially sensitive

Interactive voice

Videoconferencing

- Causes of degraded performance

  Congestion

  Convergence

  Peak traffic load

  Link speed & capacity differences

➢ Set application service level objectives

Application Examples	Sensitivity		
	Delay	Jitter	Packet Loss
Interactive Voice and Video	Y	Y	Y
Streaming Video	N	Y	Y
Transactional / Interactive	Y	N	N
Bulk Data Email File Transfer	N	N	N

Why Enable QoS? HA, Security and QoS Are Interdependent Technologies

QoS

- Enables VoIP and IP telephony
- Drives productivity by enhancing service-levels to mission-critical applications
- Cuts costs by bandwidth optimization
- Helps maintain network availability in the event of DoS/ worm attacks

QoS Service Models

- These are global, high level framework describing how QoS can be applied in a network.

- Three services models:

  Best Effort

  Integrated Services

  Differentiated Services

QoS Model #1: Best Effort

- First come, first served basis

- Network's behavior:

  Treats all traffic the same and on a first come, first served basis.

- Drawbacks

  Delivers data if it can, with no assurances of reliability, delay bounds, or throughput. So basically no QoS ;)

QoS Model #2: Integrated Services

- Dynamic allocation of resources

- Network's behavior:

  Applications requests a specific level of service before starting to send data.

- Drawbacks

  Requires explicit signaling through protocol (RSVP)

  Overhead in network services, scalability issues.

QoS Model #3: Differentiated Services

- Flows are aggregated at the edge of network

- Network's behavior:

  Smaller number of aggregated flows follow the behavior implemented on each hop ('Per Hop Behavior').

- Drawbacks

  Needs standardized policies at each hop to ensure end-to-end services

QoS Model #3: Differentiated Services DiffServ Architecture

- Network Boundaries: Traffic Conditioner Block

  Incoming traffic is classified and can be conditioned (metered, delayed, dropped)

  Is assigned to an aggregate flow matching a behavior. This is done by marking it with a DiffServ Code Point (DSCP).

- Network Core: Per Hop Behavior

  Traffic is forwarded/dropped according to the Per Hop Behavior corresponding to its DiffServ Code Point.

QoS Model #3: Differentiated Services Per Hop Behavior

- Defines the "Externally observable forwarding behavior" of a DiffServ node (loss percentage, delay, jitter, drop precedence)
- The DiffServ model associates the standard behavior of a participating node to the DSCP of the packets.
- Some convention are used to ensure consistent usage of DSCP values across networks.
- Can be split in 4 types (EF, AF, CS, default)

Quality of Service Operations
How Does It Work and Essential Elements

Classification and Marking

IDENTIFY & PRIORITIZE

Queuing and Dropping

MANAGE & SORT

Post-Queuing Operations

PROCESS & SEND

- Classification & Marking:

  The first element to a QoS policy is to classify/identify the traffic that is to be treated differently. Following classification, marking tools can set an attribute of a frame or packet to a specific value.

- Policing:

  Determine whether packets are conforming to administratively-defined traffic rates and take action accordingly. Such action could include marking, remarking or dropping a packet.

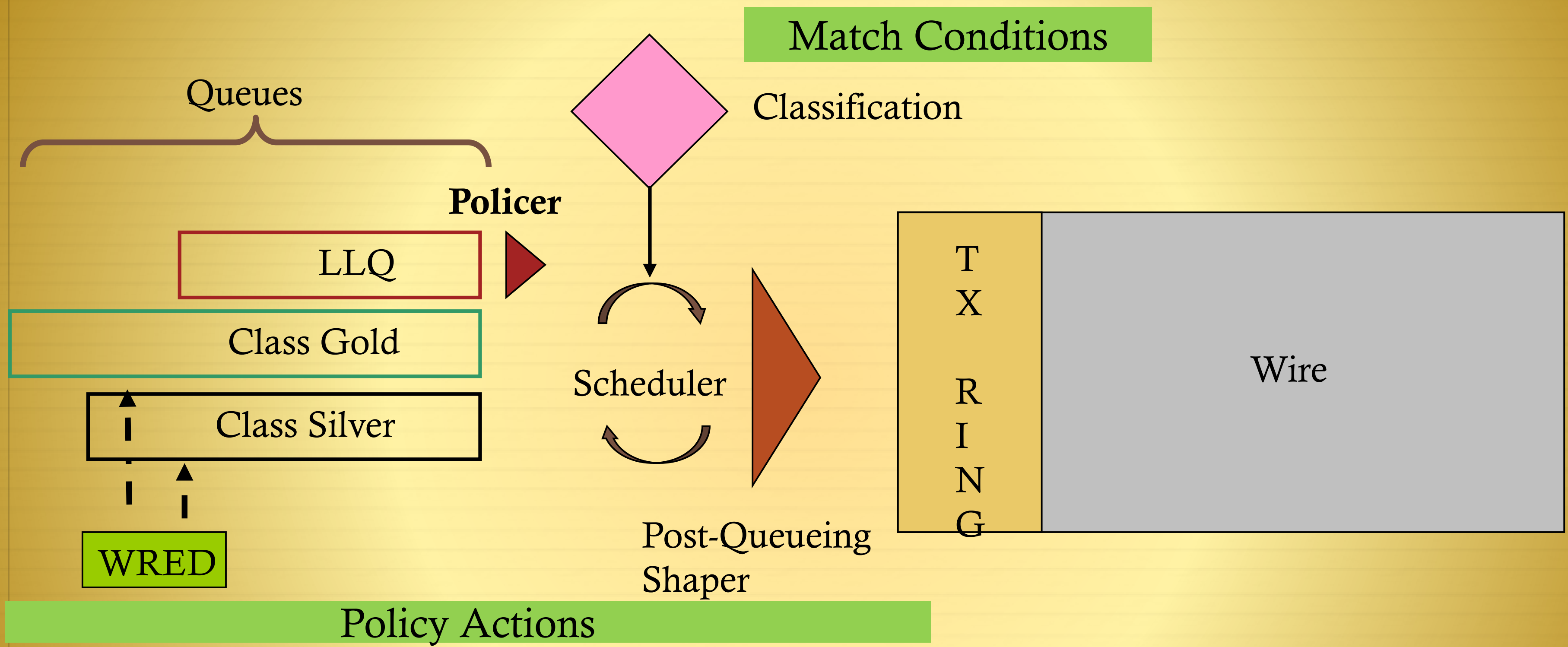
- Scheduling (including Queuing & Dropping):

  Scheduling tools determine how a frame/packet exits a device. Queuing algorithms are activated only when a device is experiencing congestion and are deactivated when the congestion clears.

- Link Specific Mechanisms (Shaping, Fragmentation, Compression, Tx Ring)

  Offers network administrators tools to optimize link utilization

Cisco IOS QoS Behavioral Model



Classification	Pre-Queuing	Queuing and Scheduling	Post-Queuing
Classify Traffic	Immediate Actions	Congestion Management and Avoidance	Link Efficiency Mechanisms

How Is QoS Optimally Deployed?

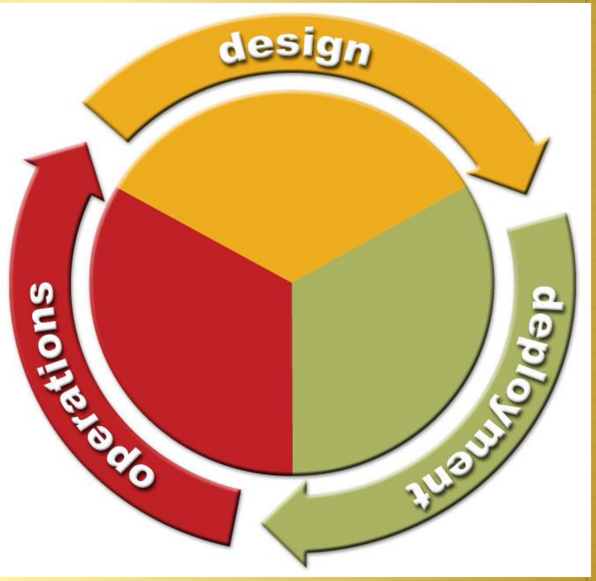
1. Strategically define the business objectives to be achieved via QoS

2. Analyze the service-level requirements of the various traffic classes to be provisioned for

3. Design and test the QoS policies prior to production-network rollout

4. Roll-out the tested QoS designs to the production-network in phases, during scheduled downtime

5. Monitor service levels to ensure that the QoS objectives are being met

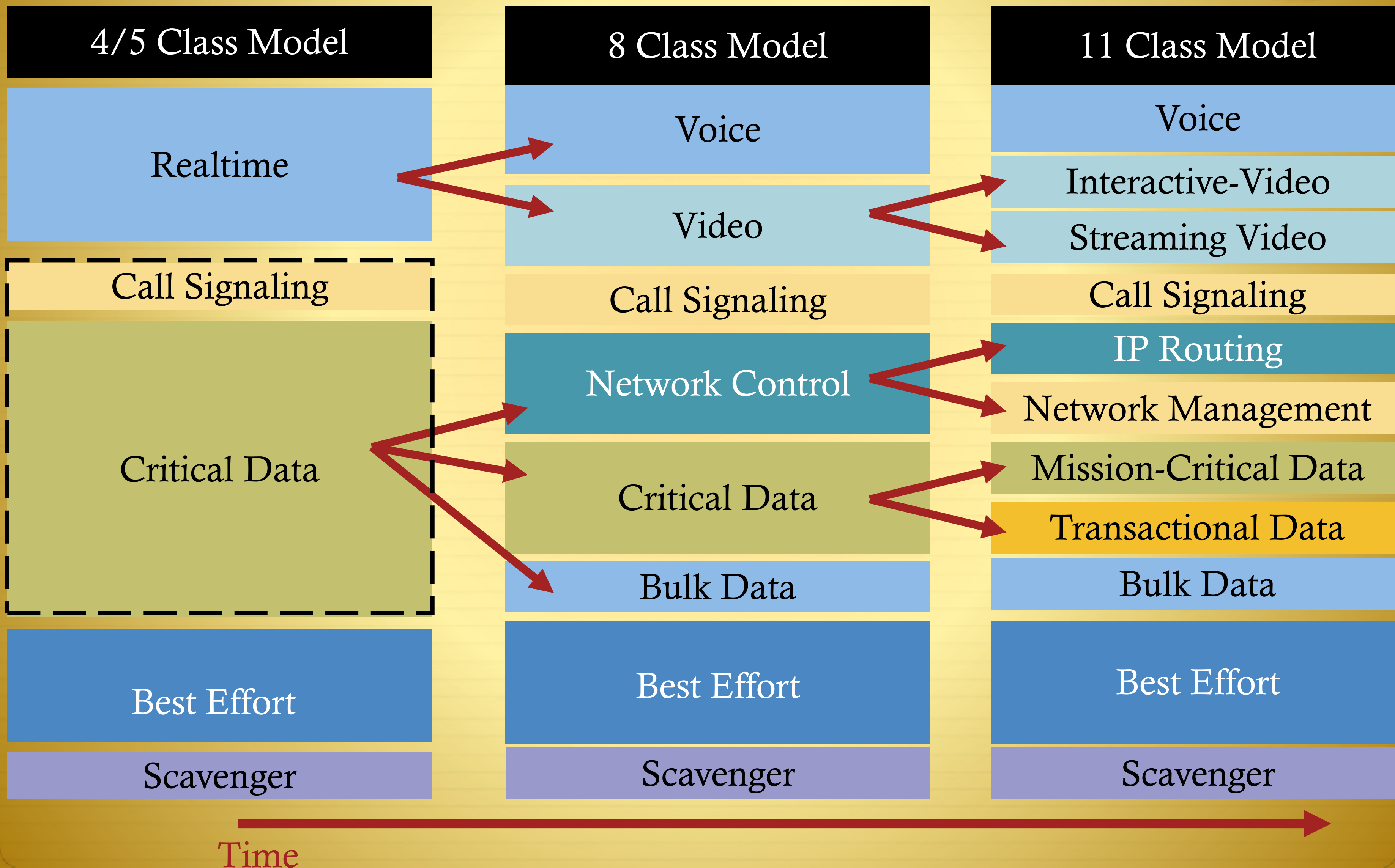


General QoS Design Principles
Start with the Objectives, Not the Tools

- Clearly define the organizational objectives
  - Protect voice? Video? Data?
  - DoS/worm mitigation?
- Assign as few applications as possible to be treated as "mission-critical"
- Seek executive endorsement of the QoS objectives prior to design and deployment
- Determine how many classes of traffic are required to meet the organizational objectives
  - More classes = more granular service-guarantees

How Many Classes of Service Do I Need?

Example Strategy for Expanding the Number of Classes of Service over Time



IPv4 ToS and IPv6 Traffic Class

- IPv4 uses 8-bit Type of Service (ToS) field
- IPv6 uses 8-bit Traffic Class field

IPv4 Header

Version	**IHL**	**ToS**	**Length**
Identification		Flags	Fragment Off.
TTL	Protocol	Header Checksum	
Source Address			
Destination Address			
Options			Padding

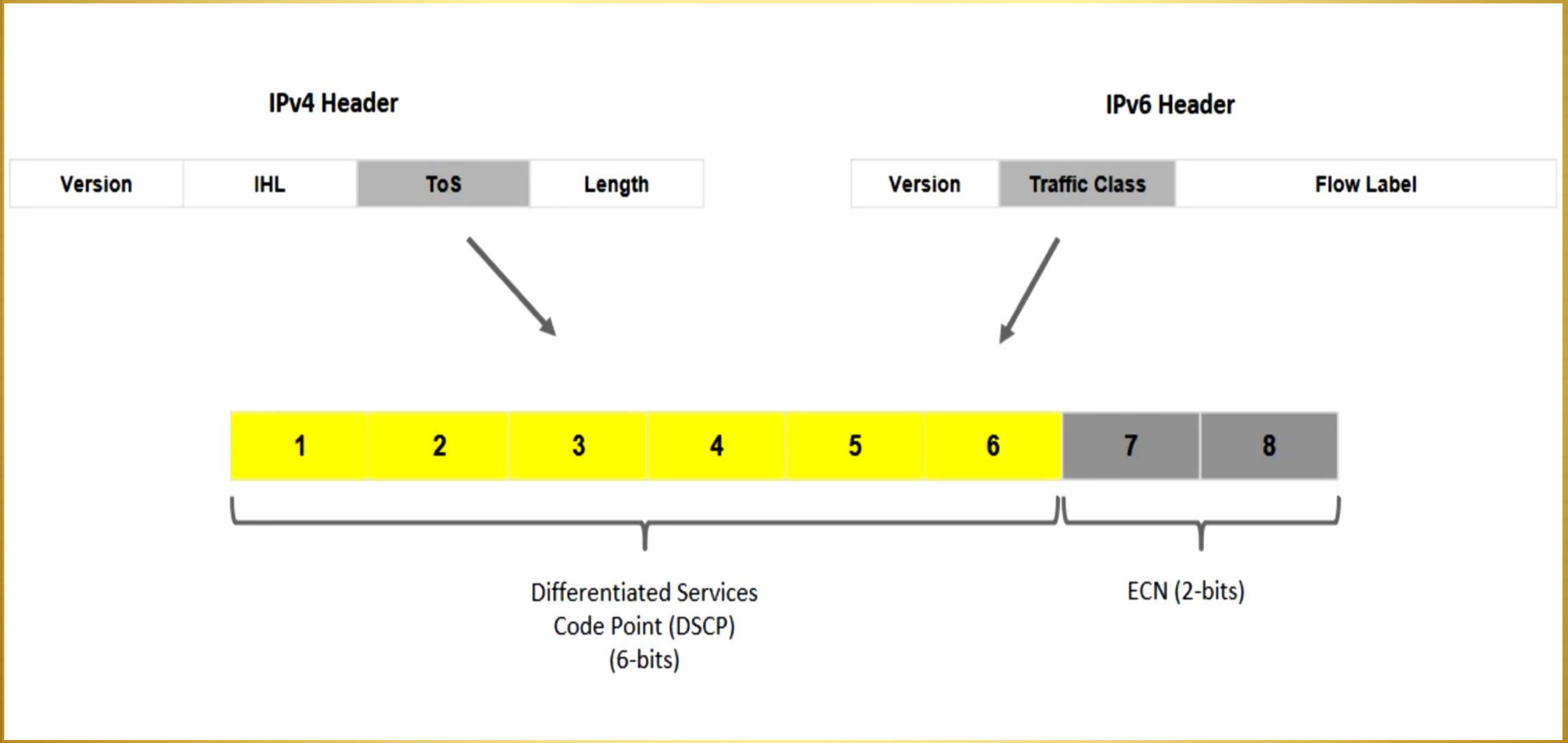
IPv6 Header

Version	**Traffic Class**	**Flow Label**	
Payload		Next Header	Hop Limit
Source Address			
Destination Address			

IP Precedence vs DSCP

Bits							
1	2	3	4	5	6	7	8
IP Precedence							
DSCP						ECN	

Example Mapping of Tos and Traffic Class with DSCP



IP Precedence Marking

Marking	Binary	Service Level
0	000	Routine
1	001	Priority
2	010	Immediate
3	011	Flash
4	100	Flash Override
5	101	Critical
6	110	Internetwork Control
7	111	Network Control

DSCP and CS

Classification DSCP	Classification CS	Application Type
DSCP 46 (EF)	CS 5	Voice Bearer
DSCP 32	CS 4	Streaming Video
DSCP 26 (Previously marked as AF31)	CS 3	Voice/Call Signaling
DSCP 8	CS 2	Network Management
DSCP 0	CS 1	Scavenger

The Assured Forwarding Classes

PHB	Low Drop Preference	Medium Drop Preference	High Drop Preference
Class 1	AF11 (10)	AF12 (12)	AF13 (14)
Class 2	AF21 (18)	AF22 (20)	AF23 (22)
Class 3	AF31 (26)	AF32 (28)	AF33 (32)
Class 4	AF41 (34)	AF42 (36)	AF43 (38)

Precedence/DSCP

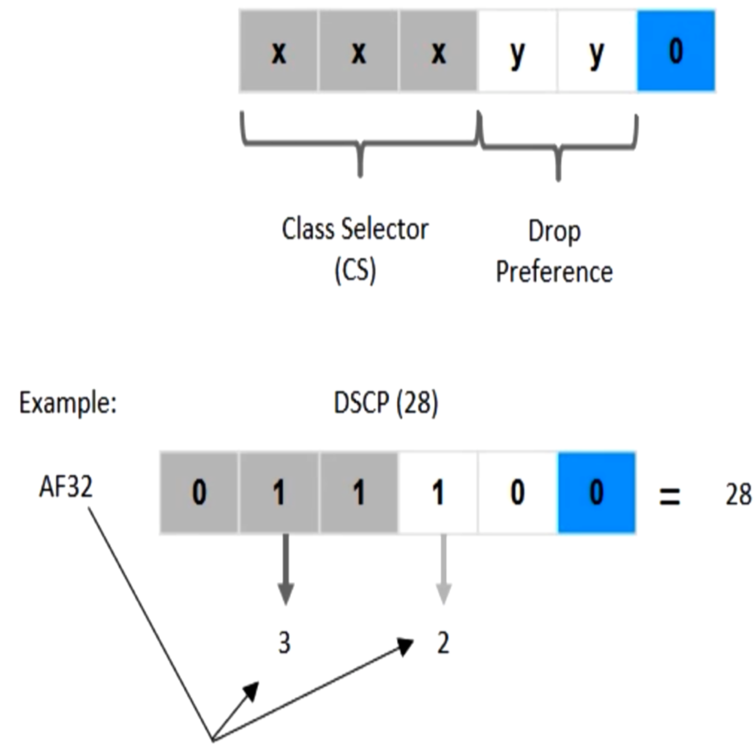
	Binary	DSCP	Prec.
56	111000	Reserved	7
48	110000	Reserved	6
46	101110	EF	5
32	100000	CS4	4
34	100010	AF41	
36	100100	AF42	
38	100110	AF43	
24	011000	CS3	3
26	011010	AF31	
28	011100	AF32	
30	011110	AF33	
16	010000	CS2	2
18	010010	AF21	
20	010100	AF22	
22	010110	AF23	
8	001000	CS1	1
10	001010	AF11	
12	001100	AF12	
14	001110	AF13	
0	000000	BE	0

Example DSCP and CS

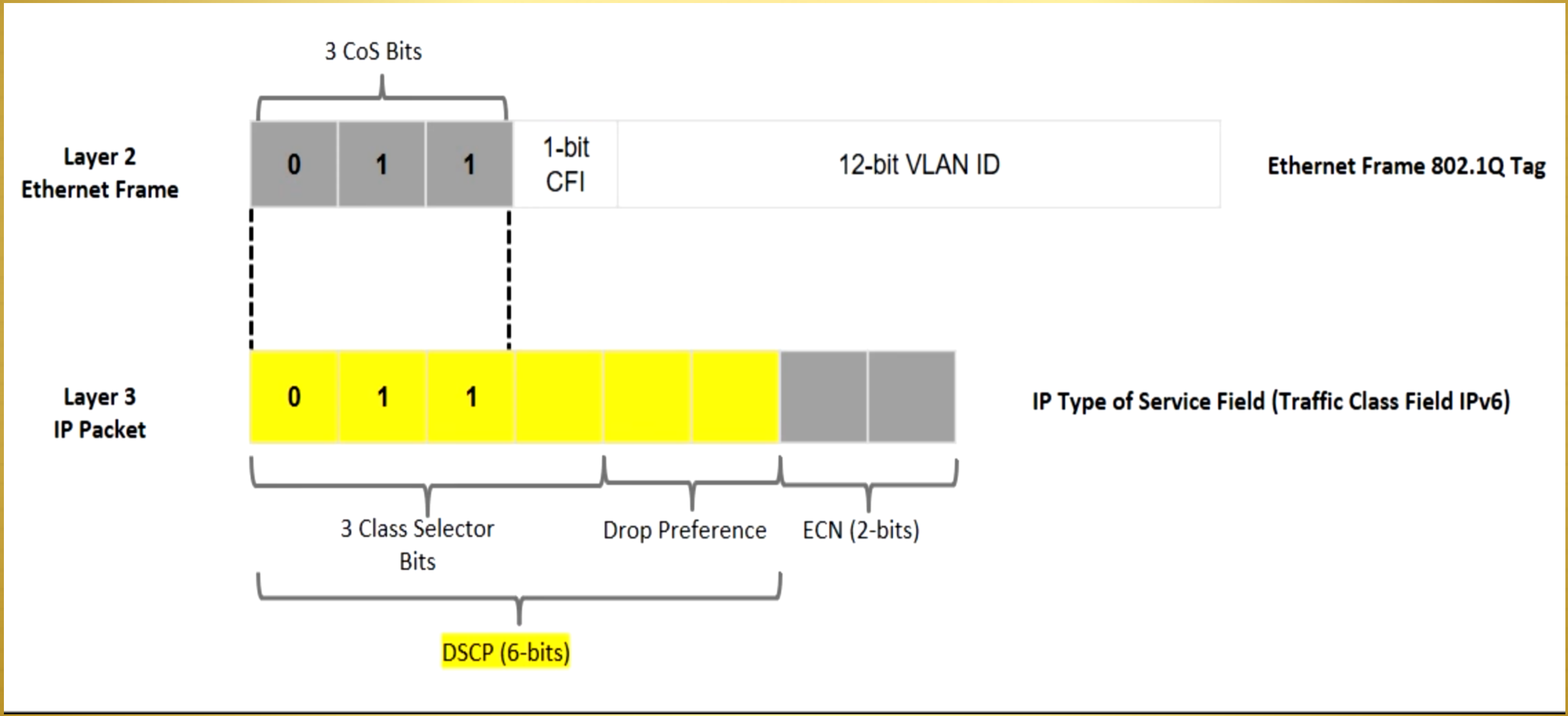
Best Queue ↑ Worst Queue

	Low Drop	Medium Drop	High Drop
Class 4	AF41 (34)	AF42 (36)	AF43 (38)
Class 3	AF31 (26)	**AF32 (28)**	AF33 (30)
Class 2	AF21 (18)	AF22 (20)	AF23 (22)
Class 1	AF11 (10)	AF12 (12)	AF13 (14)

Best Drop ← Worst Drop

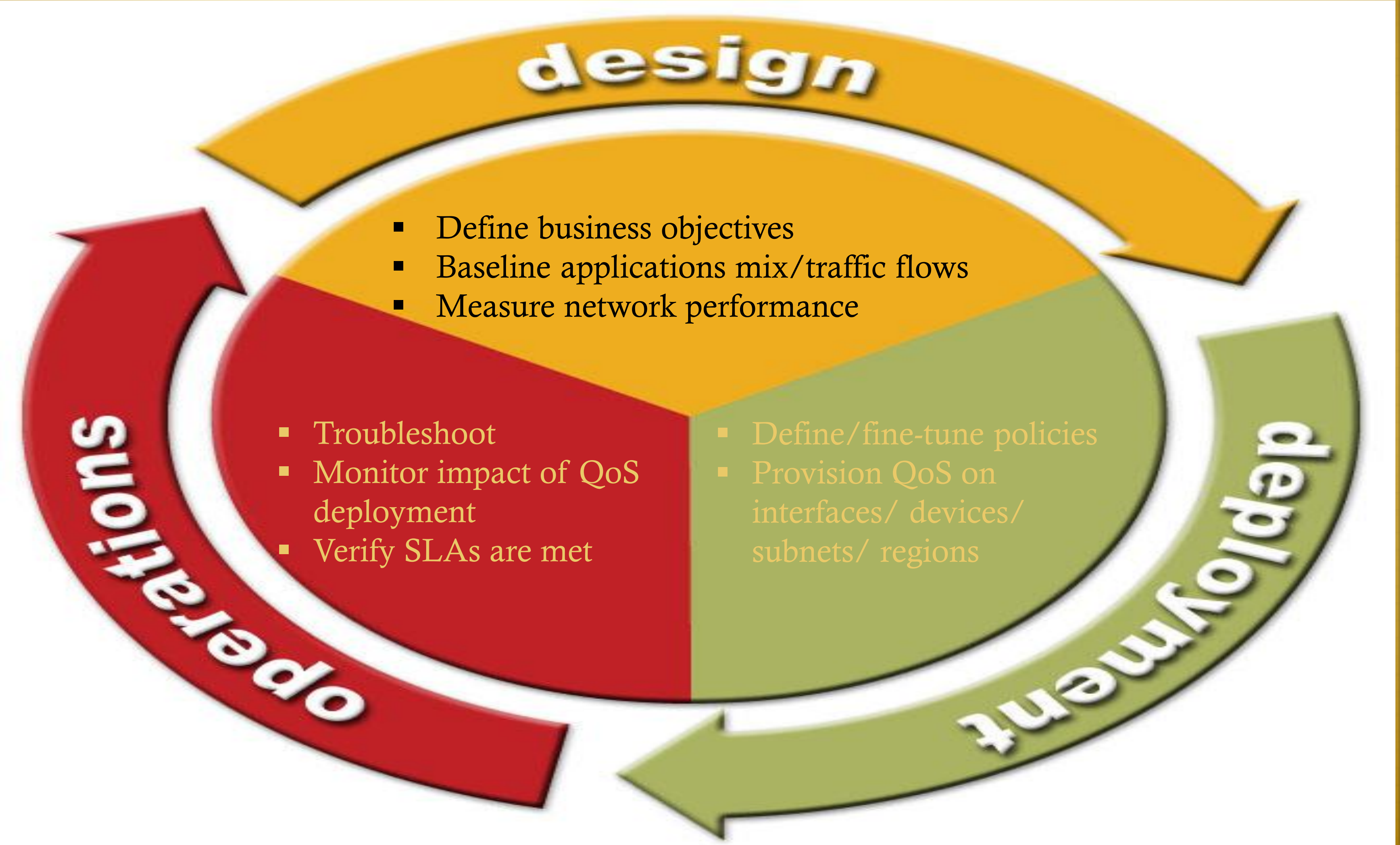


Layer 2 Marking and Layer 3 Marking



The Solution

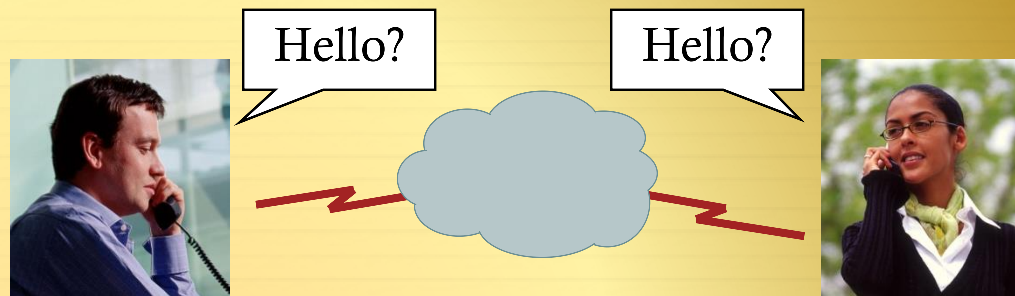
QoS Requires Lifecycle Management

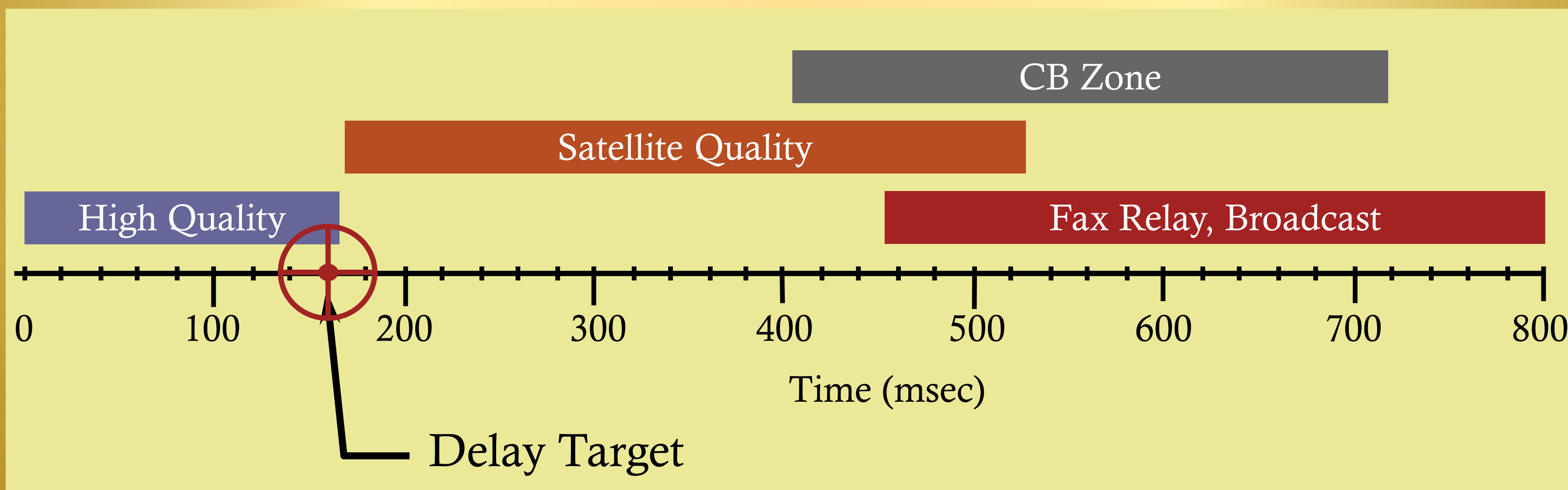


QOS for convergence

Voice QoS Requirements
End-to-End Latency

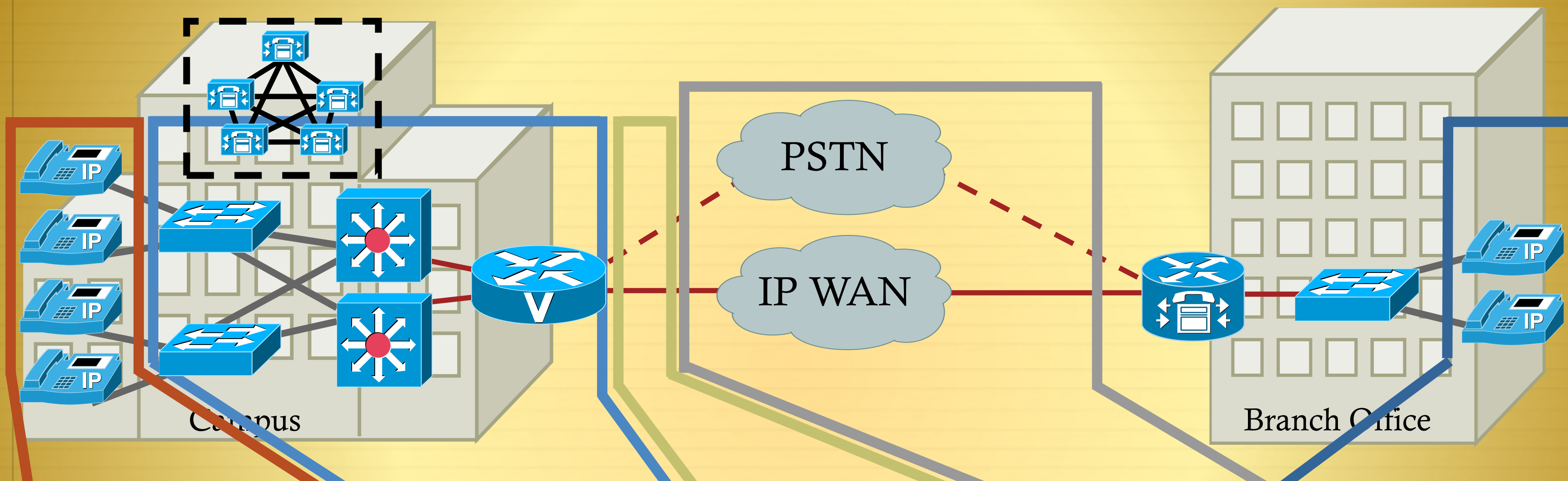
Avoid the
"Human Ethernet"





ITU's G.114 Recommendation: ≤ 150msec One-Way Delay

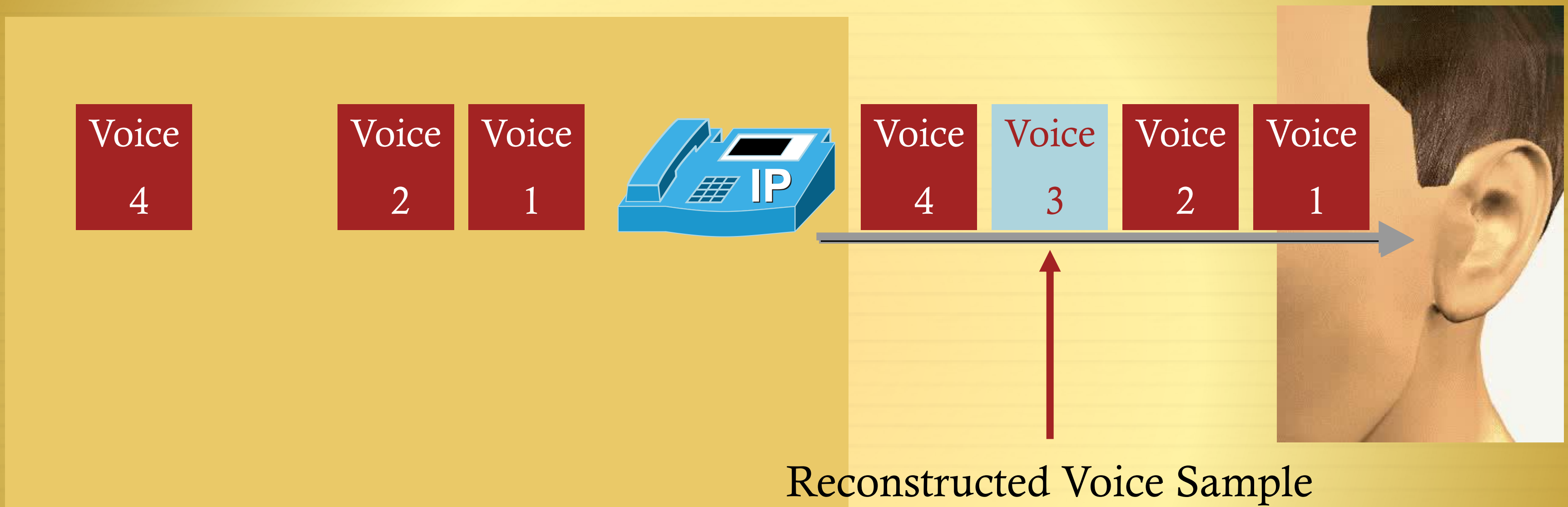
Voice QoS Requirements
Elements That Affect Latency and Jitter



CODEC	Queuing	Serialization	Propagation and Network	Jitter Buffer
G.729A: 25 ms	Variable	Variable	Fixed (3.3 µs/Km) + Network Delay (Variable)	20–50 ms

End-to-End Delay (Must Be ≤ 150 ms)

Voice QoS Requirements
Packet Loss Limitations



- Cisco DSP codecs can use predictor algorithms to compensate for a single lost packet in a row
- Two lost packets in a row will cause an audible clip in the conversation

Voice QoS Requirements
Provisioning for Voice

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

One-Way Requirements

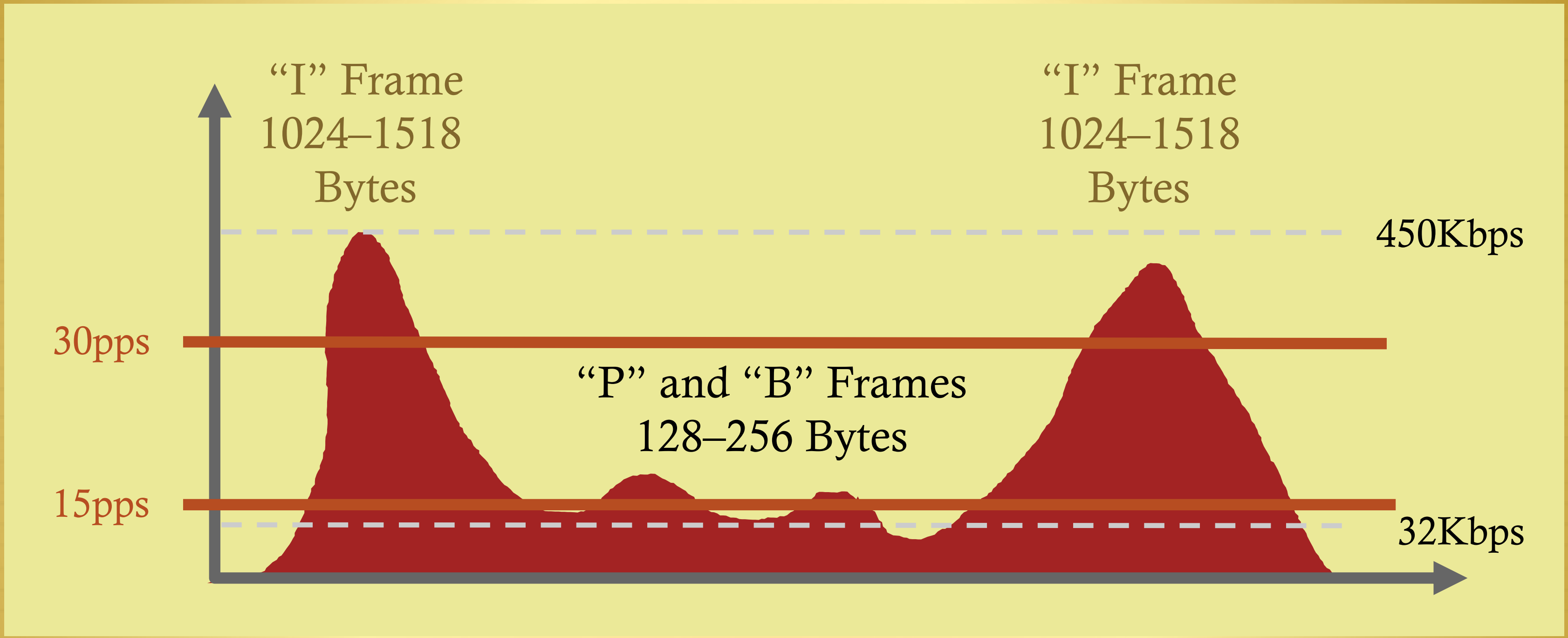
- 17–106 kbps guaranteed priority bandwidth per call
- 150 bps (+ layer 2 overhead) guaranteed bandwidth for voice-control traffic per call
- CAC must be enabled

Voice

- Smooth
- Benign
- Drop sensitive
- Delay sensitive
- UDP priority

Video QoS Requirements

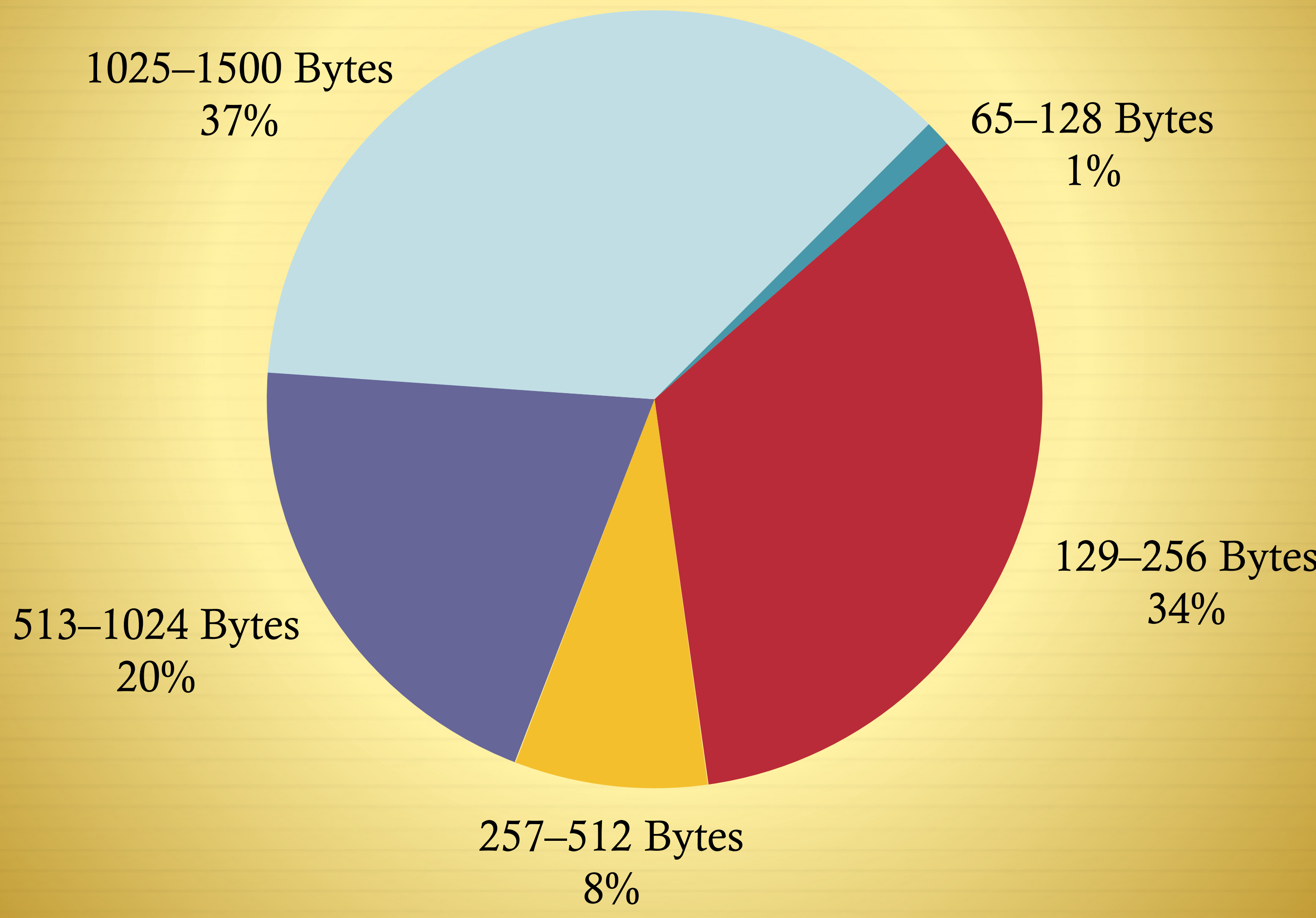
Video Conferencing Traffic Example (384 kbps)



- "I" frame is a full sample of the video
- "P" and "B" frames use quantization via motion vectors and prediction algorithms

Video QoS Requirements

Video Conferencing Traffic Packet Size Breakdown

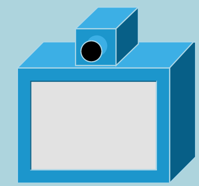


Video QoS Requirements Provisioning for Interactive Video

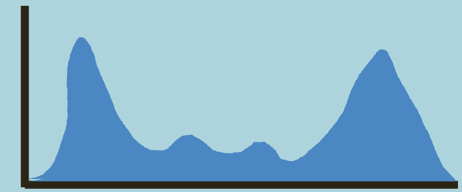
- Latency ≤ 150 ms
- Jitter ≤ 30 ms
- Loss ≤ 1%

One-Way Requirements

- Minimum priority bandwidth guarantee required is
  - Video-stream + 10–20%
  - e.g., a 384 kbps stream could require up to 460 kbps of priority bandwidth
- CAC must be enabled

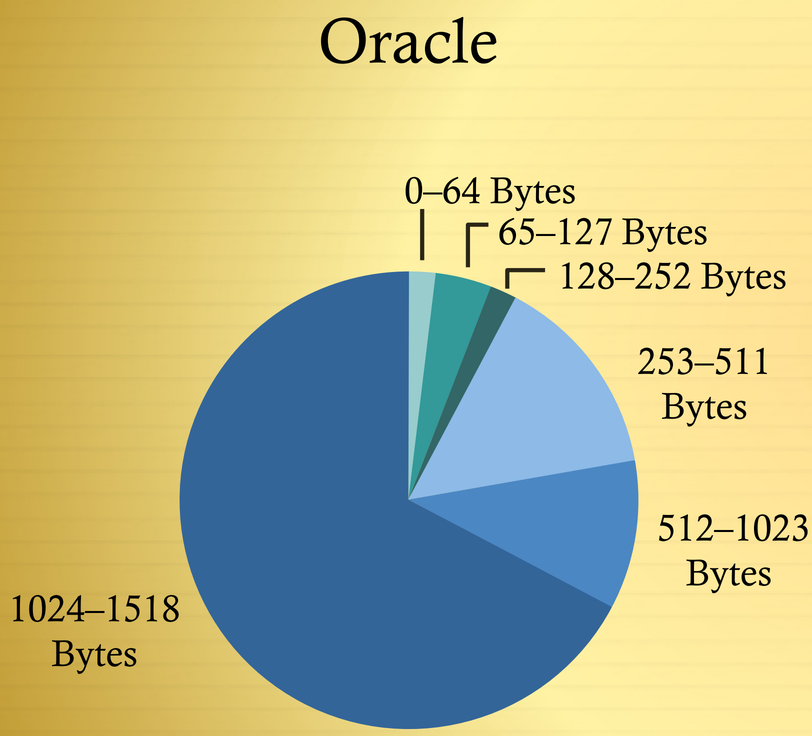


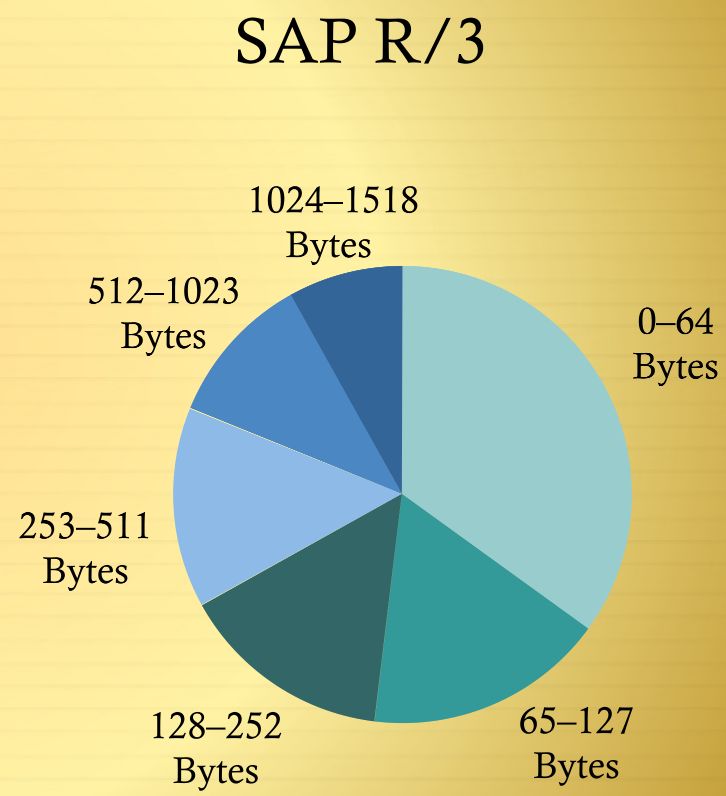
Video



- Bursty
- Drop sensitive
- Delay sensitive
- UDP priority

Data QoS Requirements Application Differences



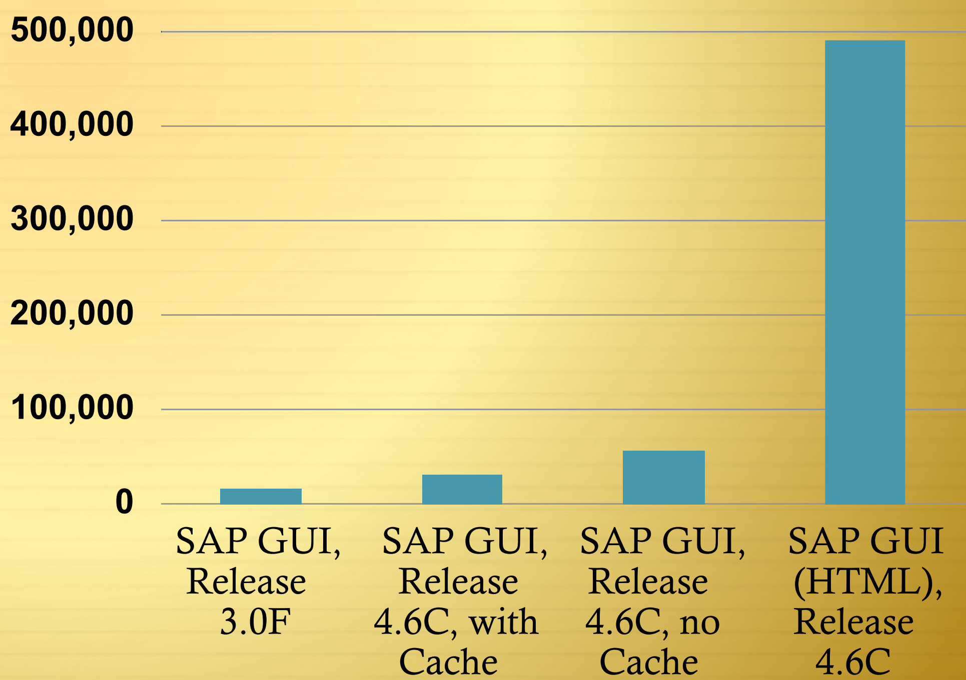


Data QoS Requirements
Version Differences

Same Transaction Takes Over 35 Times More Traffic from One Version of an Application to Another

SAP Sales Order Entry Transaction

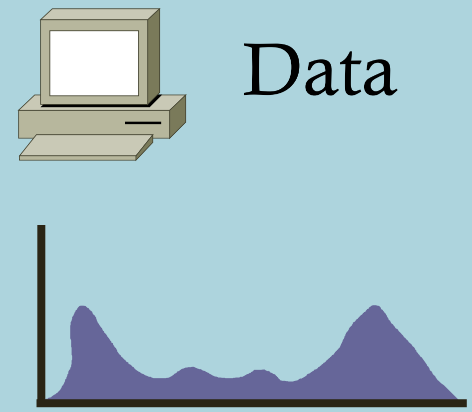
Client Version	VA01 # of Bytes
SAP GUI Release 3.0 F	14,000
SAP GUI Release 4.6C, No Cache	57,000
SAP GUI Release 4.6C, with Cache	33,000
SAP GUI for HTML, Release 4.6C	490,000



Data QoS Requirements
Provisioning for Data

- Different applications have different traffic characteristics
- Different versions of the same application can have different traffic characteristics
- Classify data into four/five data classes model
  - Mission-critical apps
  - Transactional/interactive apps
  - Bulk data apps
  - Best effort apps
  - Optional: Scavenger apps

Data



- Smooth/bursty
- Benign/greedy
- Drop insensitive
- Delay insensitive
- TCP retransmits

Data QoS Requirements Provisioning for Data (Cont.)

- Use four/five main traffic classes
  - Mission-critical apps—business-critical client-server applications
  - Transactional/interactive apps—foreground apps: client-server apps or interactive applications
  - Bulk data apps—background apps: FTP, e-mail, backups, content distribution
  - Best effort apps—(default class)
  - Optional: Scavenger apps—peer-to-peer apps, gaming traffic
- Additional optional data classes include internetwork-control (routing) and network-management
- Most apps fall under best-effort, make sure that adequate bandwidth is provisioned for this default class

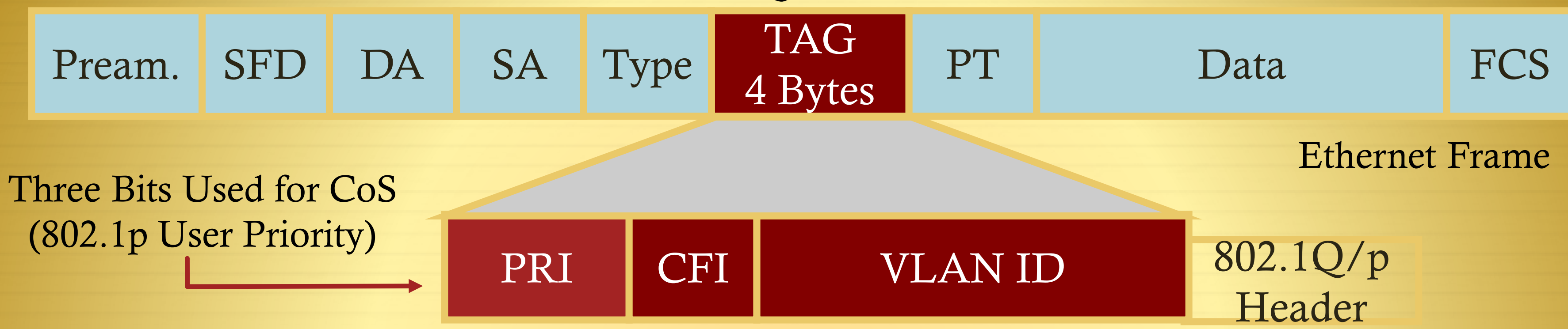
Scavenger-Class
What Is the Scavenger Class?

- The Scavenger class is an Internet 2 draft specification for a "less than best effort" service
- There is an implied "good faith" commitment for the "best effort" traffic class
  - It is generally assumed that at least some network resources will be available for the default class
- Scavenger class markings can be used to distinguish out-of-profile/abnormal traffic flows from in-profile/normal flows
  - The Scavenger class marking is CS1, DSCP 8
- Scavenger traffic is assigned a "less-than-best effort" queuing treatment whenever congestion occurs

QoS Technologies Review Classification Tools

- **Layer 1 (L1) parameters**
  - Physical interface, subinterface, PVC or port
- **Layer 2 (L2) parameters**
  - MAC address, 802.1Q/p class of service (CoS) bits, VLAN identification, experimental bits (MPLS EXP), ATM cell loss priority (CLP) and Frame Relay discard eligible (DE) bits
- **Layer 3 (L3) parameters**
  - IP Precedence, DiffServ code point (DSCP), source/destination IP address
- **Layer 4 (L4) parameters**
  - TCP or User Datagram Protocol (UDP) ports
- **Layer 7 (L7) parameters**
  - Application signatures and uniform resource locators (URLs) in packet headers or payload

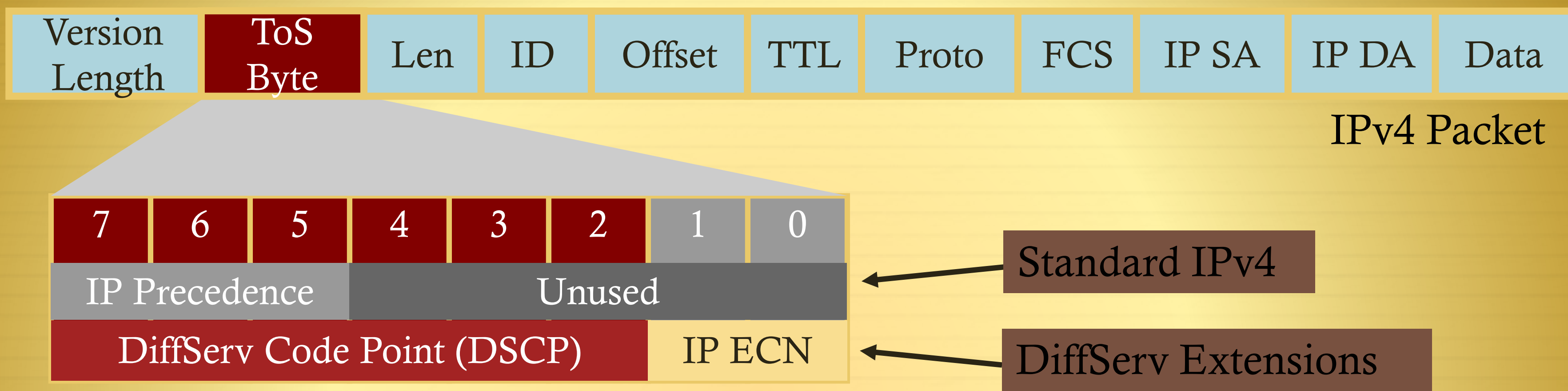
Classification Tools
Ethernet 802.1Q Class of Service



- 802.1p user priority field also called Class of Service (CoS)
- Different types of traffic are assigned different CoS values
- CoS 6 and 7 are reserved for network use

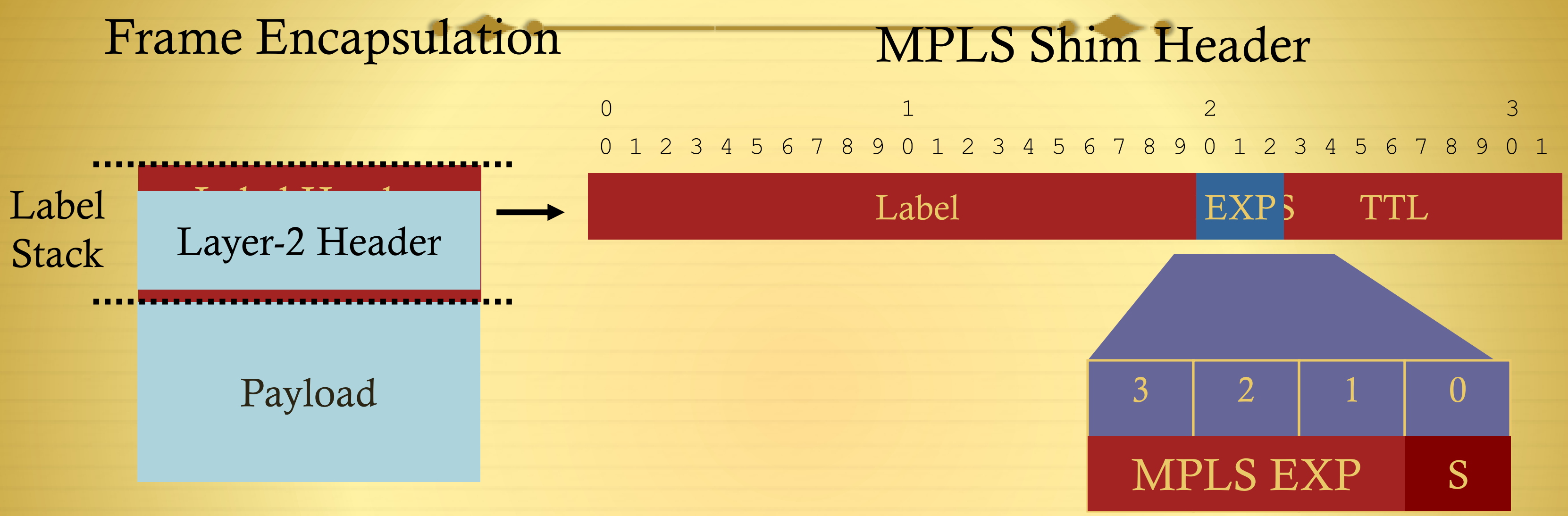
CoS	Application
7	Reserved
6	Routing
5	Voice
4	Video
3	Call Signaling
2	Critical Data
1	Bulk Data
0	Best Effort Data

Classification Tools
IP Precedence and DiffServ Code Points



- 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

Classification Tools
MPLS EXP Bits



- Packet class and drop precedence inferred from EXP (three-bit) field
- RFC3270 does not recommend specific EXP values for DiffServ PHB (EF/AF/DF)
- Used for frame-based MPLS

Classification Tools
DSCP Per-Hop Behaviors

- IETF RFCs have defined special keywords, called Per-Hop Behaviors, for specific DSCP markings
- Can be split in 4 types:

1. Default PHB: 0
2. Class Selector PHB: IP Precedence
3. Assured Forwarding PHB: AF
4. Expedite Forwarding PHB: EF

Classification Tools
DSCP Per-Hop Behaviors Types

1. Default PHB BE: Best Effort or Default Marking Value (RFC2474)

   DSCP Value 000000, maps to IP Precedence 0

2. CSx: Class Selector PHB (RFC2474)

   - Where x corresponds to the IP Precedence value (1–7)
   - (DSCP 8, 16, 24, 32, 40, 48, 56)
   - DSCP Value xxx000 maps to IP Precedence dec(xxx)

   - Values of 110000 and 111000 should always have preferential treatment to preserve common values of routing traffic (precedence 6 and 7)

Classification Tools
DSCP Per-Hop Behaviors Types

- AFxy: Assured Forwarding PHP (RFC2597)

  Where x corresponds to the IP Precedence value
  (only 1–4 are used for AF Classes) and y corresponds to the Drop Preference value (either 1 or 2 or 3) with the higher values denoting higher likelihood of dropping

  Guaranteed Bandwidth + Extra if available

  4 classes (af1, af2, af3, af4)

  3 drop probability values per class

  (DSCP 10/12/14, 18/20/22, 26/28/30, 34/36/38)

- EF: Expedite Forwarding PHB (RFC3246)

  Minimum departure rate (minimum delay)

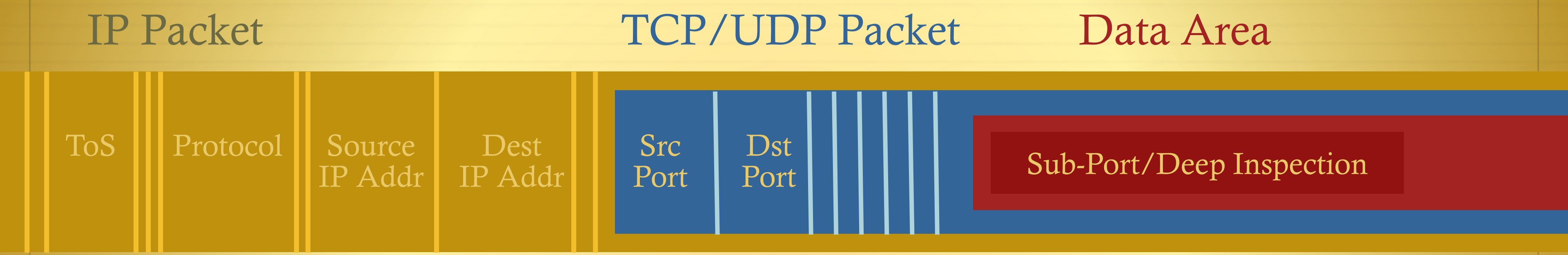
  Guaranteed Bandwidth + Drop if excess (Policed)

  DSCP Value 101110

  - (DSCP 46)

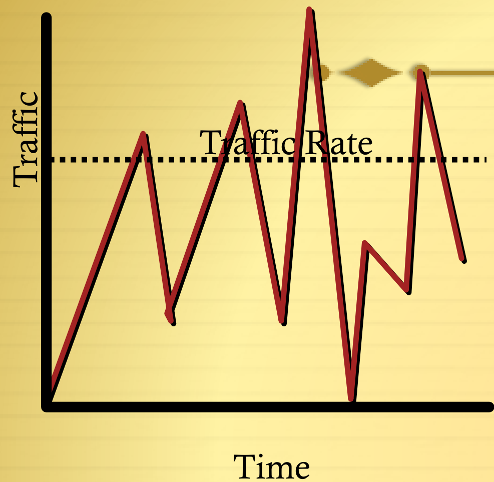
Classification Tools
Network-Based Application Recognition

Stateful and Dynamic Inspection



- Identifies over 90 applications and protocols TCP and UDP port numbers (PDLM)
  - Statically assigned
  - Dynamically assigned during connection establishment
- Non-TCP and non-UDP IP protocols
- Data packet inspection for matching values

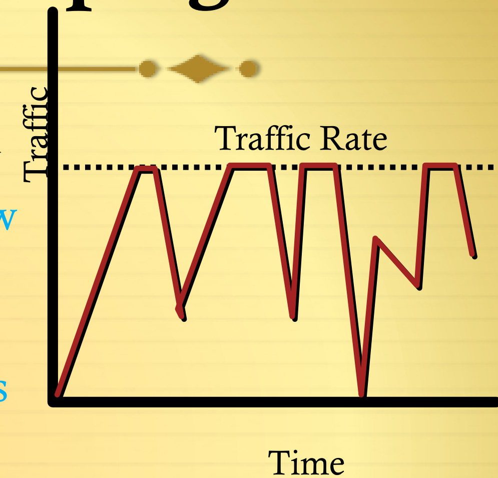
Traffic Conditioning
Policing vs Shaping

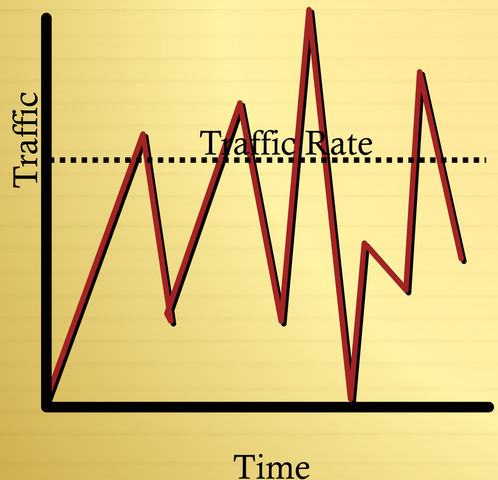


Policing

Limits traffic flow to a configured bit rate.

Drops or remarks out-of-profile packets.

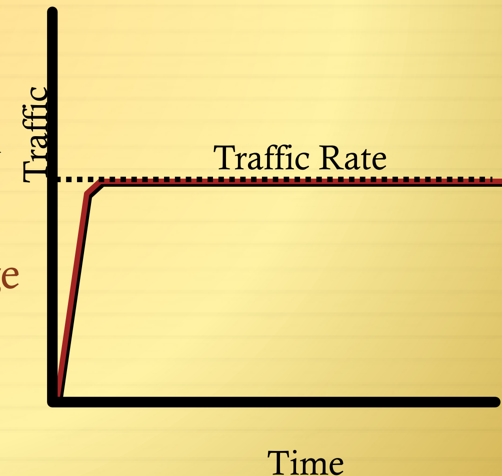




Shaping

Regulates traffic flow to an average or peak bit rate.

Commonly used where speed-mismatches exist
.



Policing Tools
Token Bucket Algorithms

- Metering engines that keep track of how much traffic can be sent to conform to the specified traffic rates
- **CIR** (Commited Information Rate)
  - The CIR is the access bit rate contracted with a service provider or the service level to be maintained.
  - specified rate at which **tokens** are granted at the beginning of some time increment (typically per second)
  - A token permits the algorithm to send a single bit (or, in some cases, a byte) of traffic.
  - i.e. if the CIR is set to 8000 bps, then 8000 tokens are placed in a "bucket" at the beginning of the time period.
- To impose CIR on interface, TDM (Time Division Multiplexing) is used: clock rate of interface not changeable to enforce policy…
  - when a rate limit (or CIR) is imposed on an interface, the limited traffic is allocated a subsecond time slice during which it can be sent.
  - i.e. if an 8-kbps CIR is imposed on a 64-kbps link, traffic can be sent for an interval of 125 ms (64,000 bps / 8000 bits).

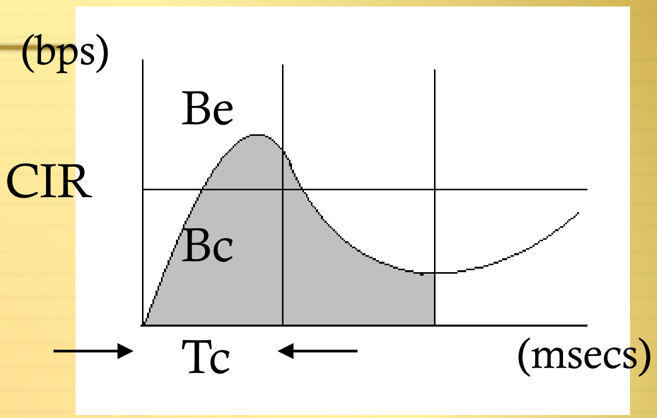
Policing Tools
Token Bucket Algorithms

- Committed Burst Size (Bc / CBS)
  - The entire amount of the CIR (8000 bits) could be sent at once, but then the algorithm would have to wait 875 ms before it could send any more data (to impose the rate limit).
  - To smooth out the flow over each second, the CIR is divided into smaller units, referred to as the committed burst (Bc), which is the sustained number of bits that can be transmitted per interval.
  - Continuing previous example:
  - if the Bc is set to 1000, each committed burst can take only 15.6 ms (1000 bits / 64,000 bps) to send traffic out the interface at the clock rate. The algorithm waits 109.4 ms (125 ms – 15.6 ms) and sends another 15.6 ms of data. This process is repeated a total of eight times during the second.

Policing Tools
Token Bucket Algorithms

- Token Bucket Algorithm:

Tc = Bc / CIR



Supported values for Tc range from 10 ms to 125 ms.

If Bc/CIR >= 125 msec, Cisco IOS will use best Tc value for stability, meaning is will round up or down the extremes.

If Bc/CIR <= 125 ms, Cisco IOS uses the Tc calculated from Bc/CIR.

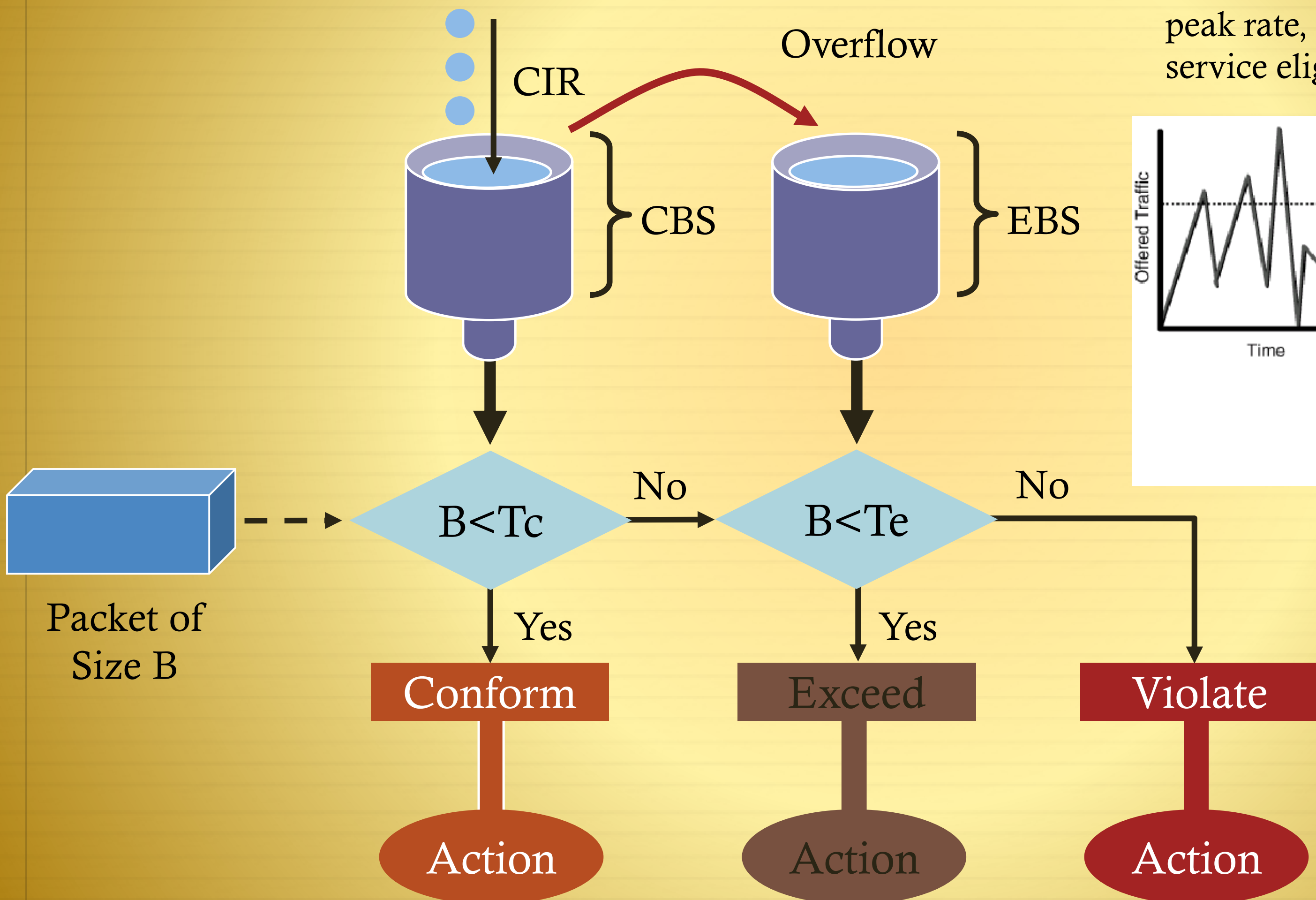
Selecting Bc Values for Data:

```
Bc = CIR/8  (where Tc = 125 msec = 1/8 sec)
```

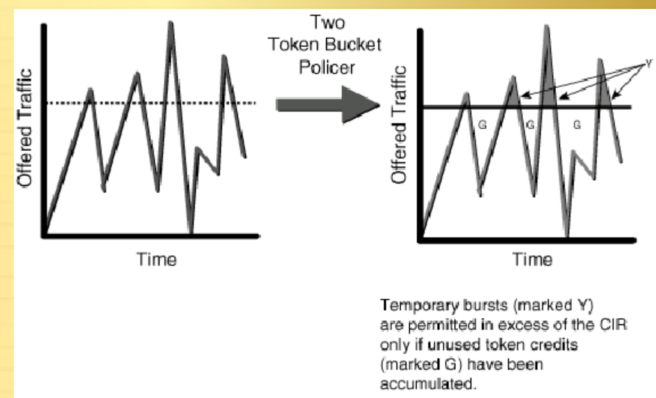
Selecting Bc values for Voice:

```
Bc = CIR/100 (where Tc = 10msec = 1/100 sec)
```

Policing Tools
RFC 2697 Single Rate Three Color Policer

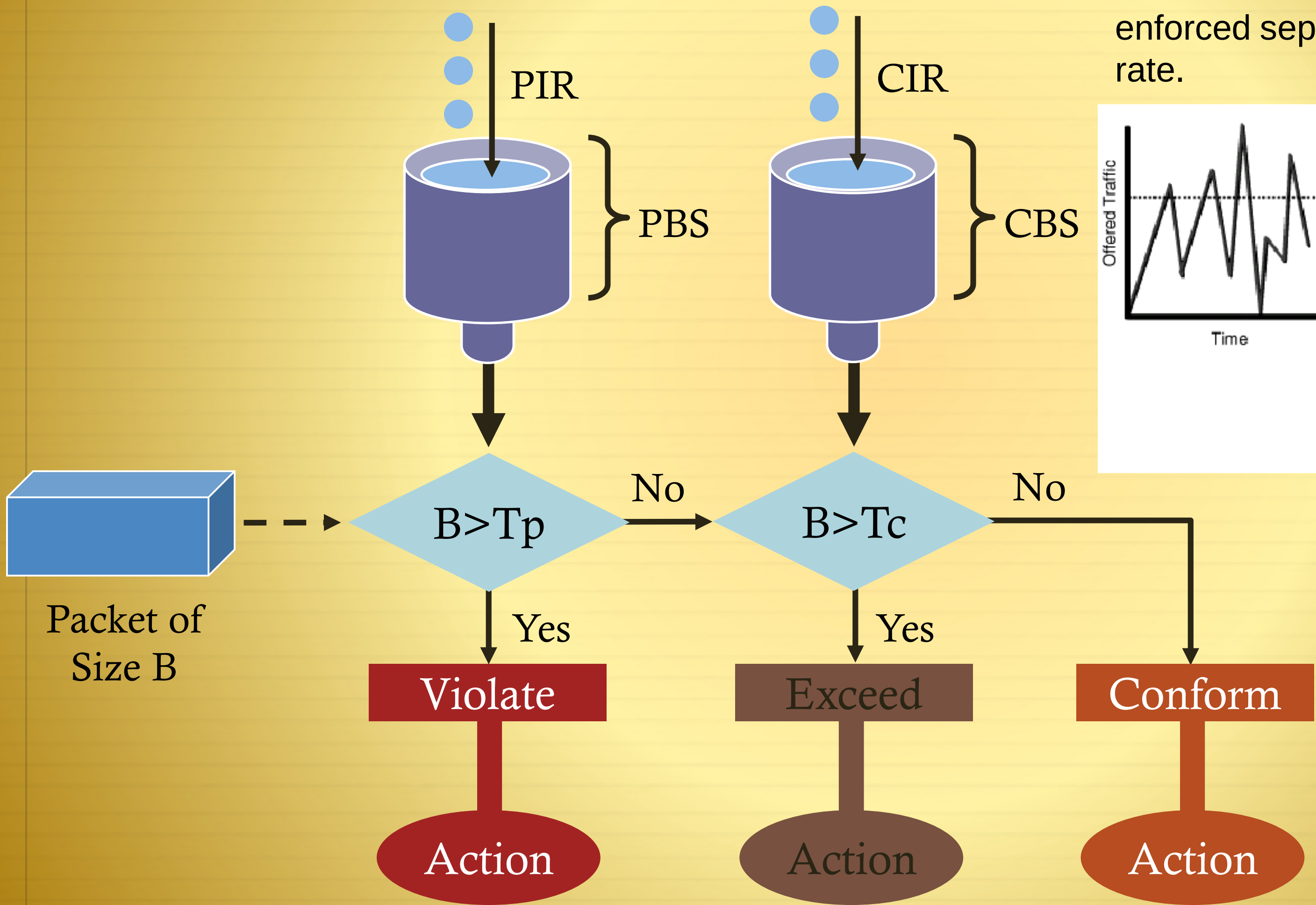


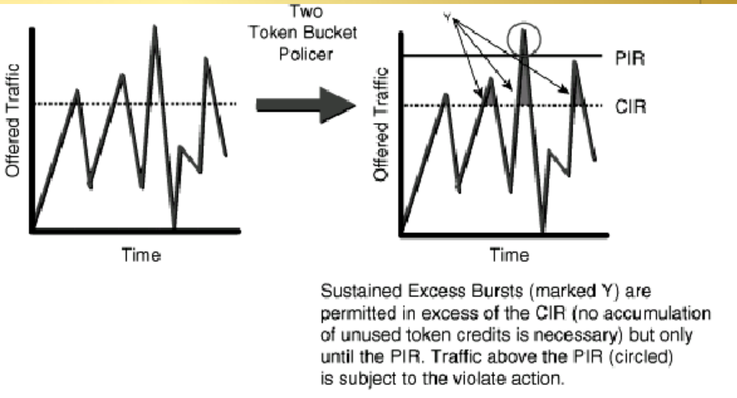
Used where only the length, not the peak rate, of the burst determines service eligibility.



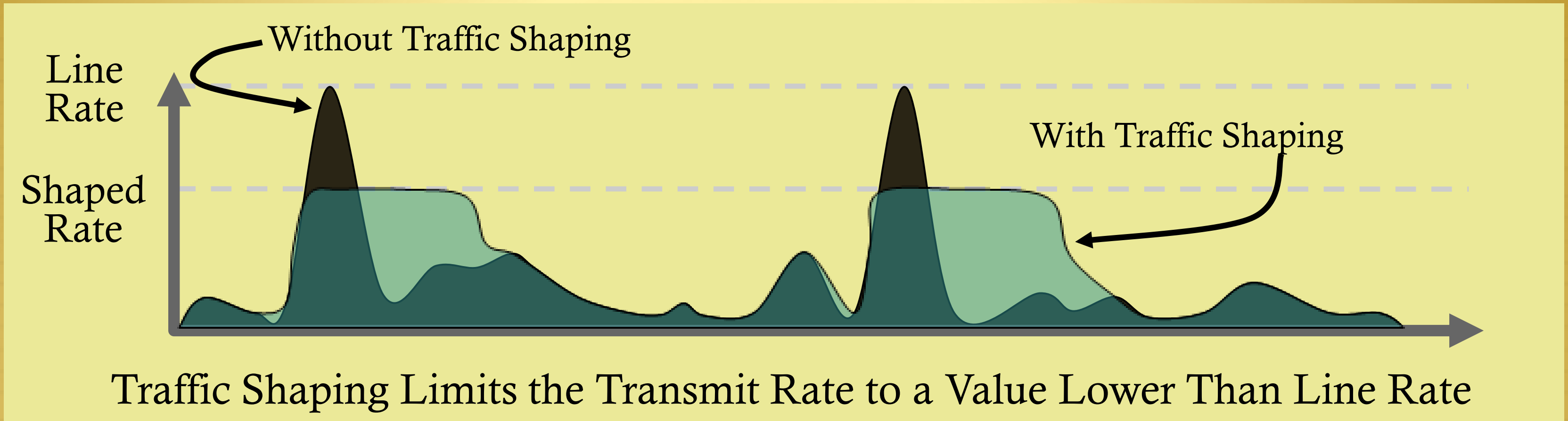
Policing Tools
RFC 2698 Two Rate Three Color Marker (trTCM)

Used where a peak rate needs to be enforced separately from a committed rate.



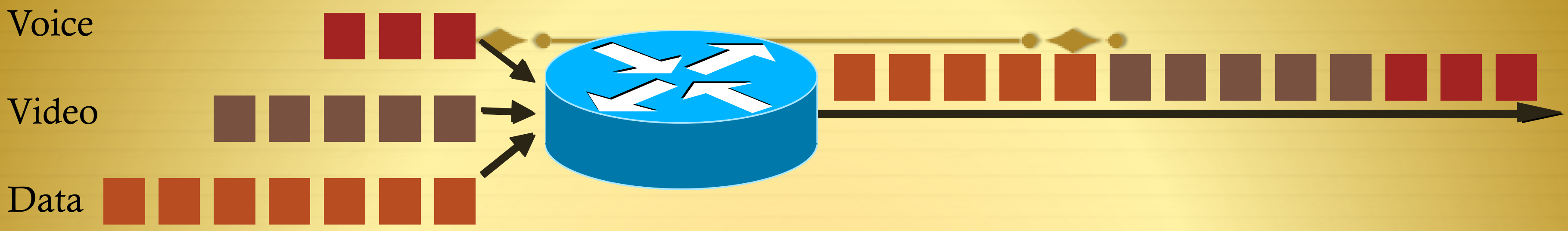


Traffic Shaping



- Policers typically drop traffic
- Shapers typically delay excess traffic, smoothing bursts and preventing unnecessary drops
- Very common on Non-Broadcast Multiple-Access (NBMA) network topologies such as Frame Relay and ATM

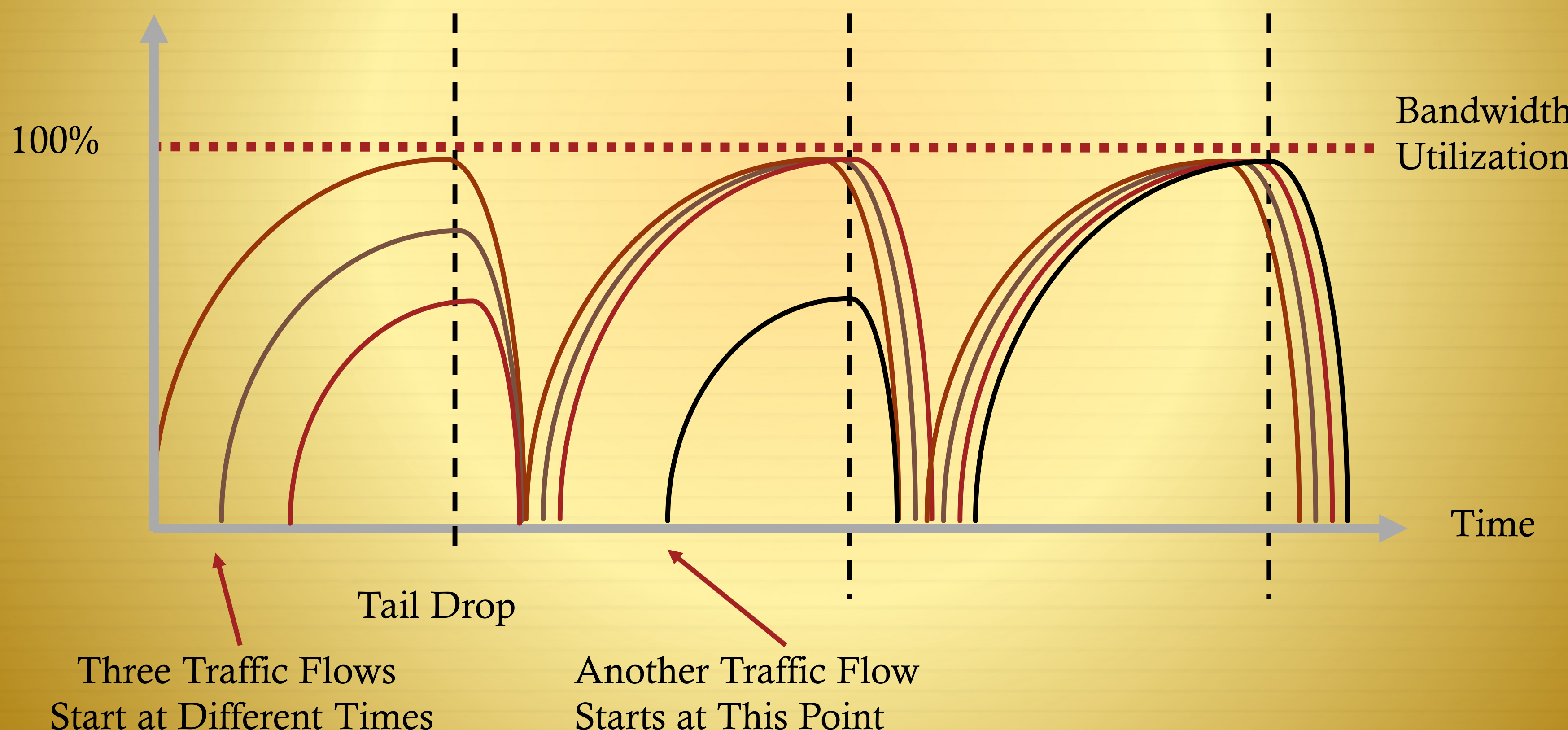
Scheduling Tools
Queuing Algorithms



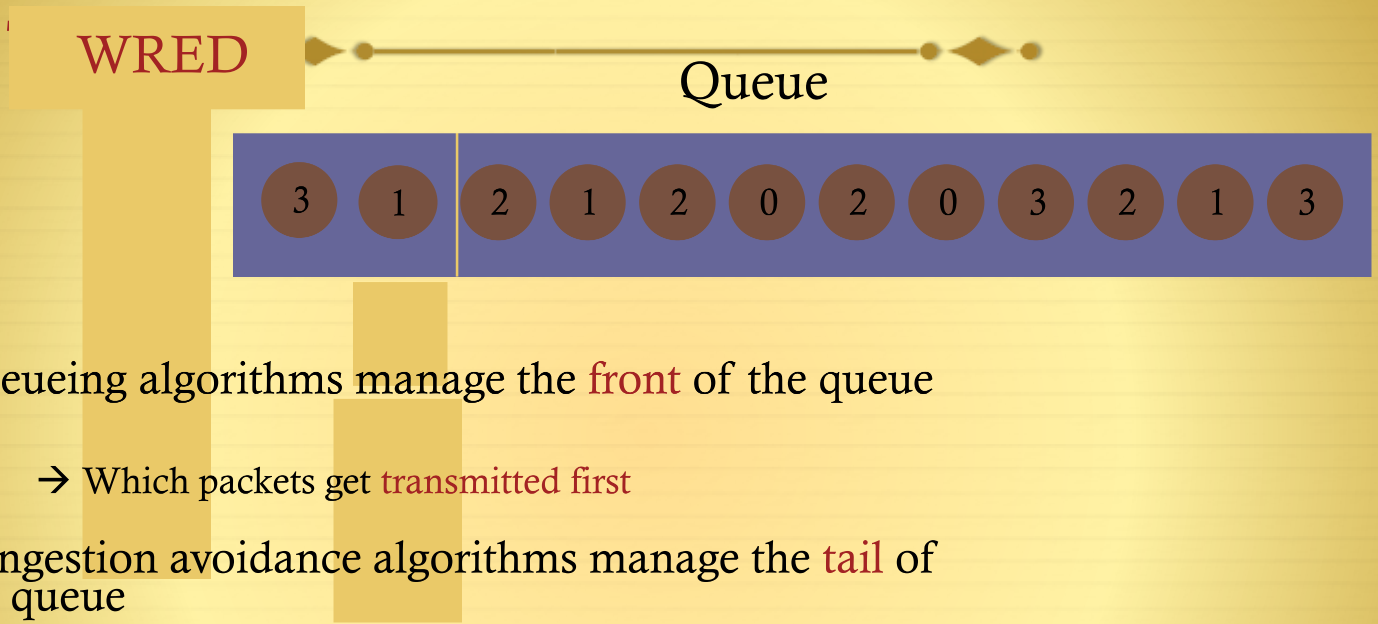
- Congestion can occur at any point in the network where there are speed mismatches
- Routers use Cisco IOS-based software queuing
  - Low-Latency Queuing (LLQ) used for highest-priority traffic (voice/video)
  - Class-Based Weighted-Fair Queuing (CBWFQ) used for guaranteeing bandwidth to data applications

TCP Global Synchronization: The Need for Congestion Avoidance

- All TCP flows synchronize in waves
- Synchronization wastes available bandwidth

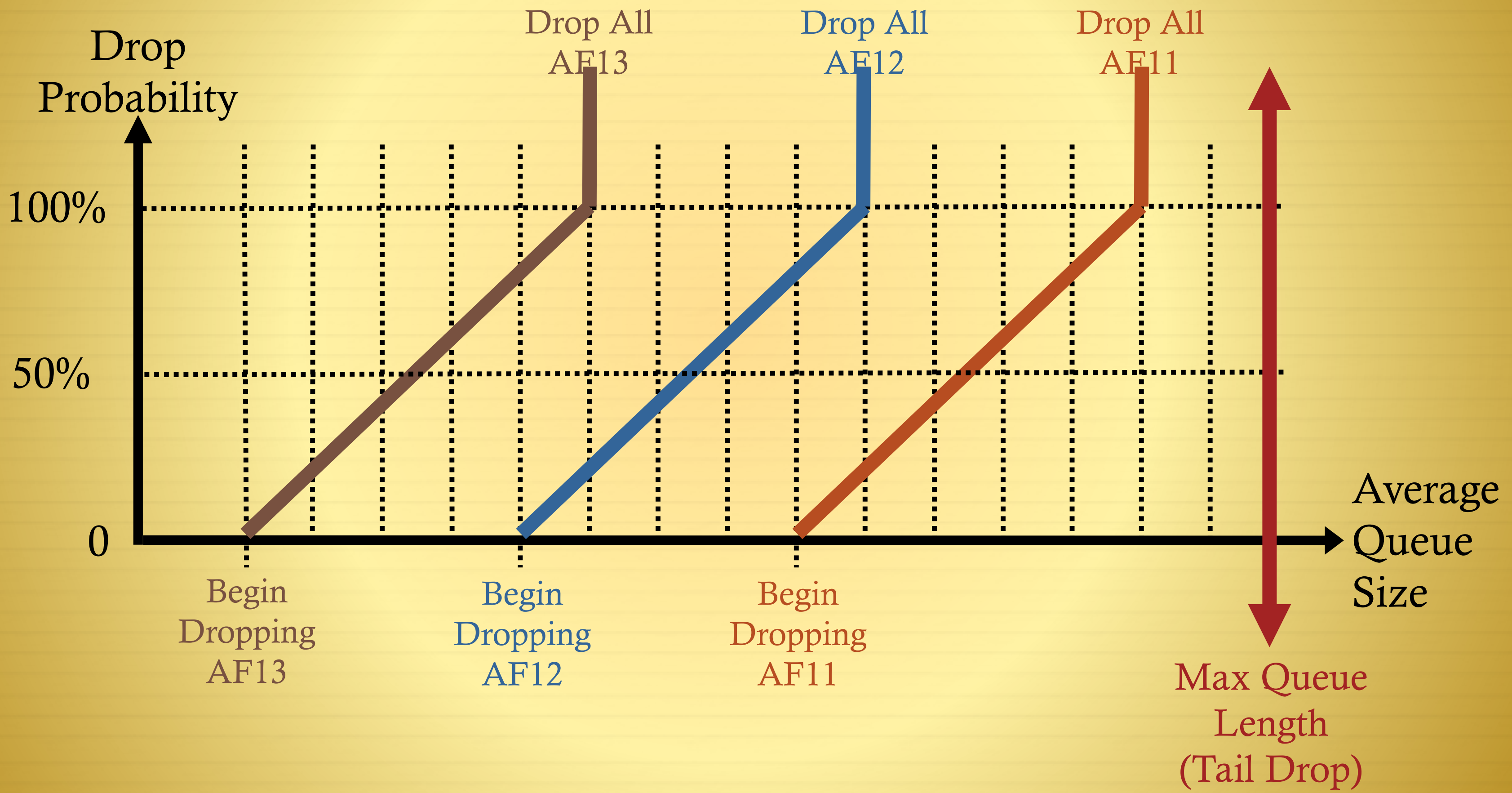


Scheduling Tools
Congestion Avoidance Algorithms



- Queueing algorithms manage the front of the queue
  - → Which packets get transmitted first
- Congestion avoidance algorithms manage the tail of the queue
  - → Which packets get dropped first when queuing buffers fill
- Weighted Random Early Detection (WRED)
  - WRED can operate in a DiffServ-compliant mode
  - → Drops packets according to their DSCP markings
  - WRED works best with TCP-based applications, like data

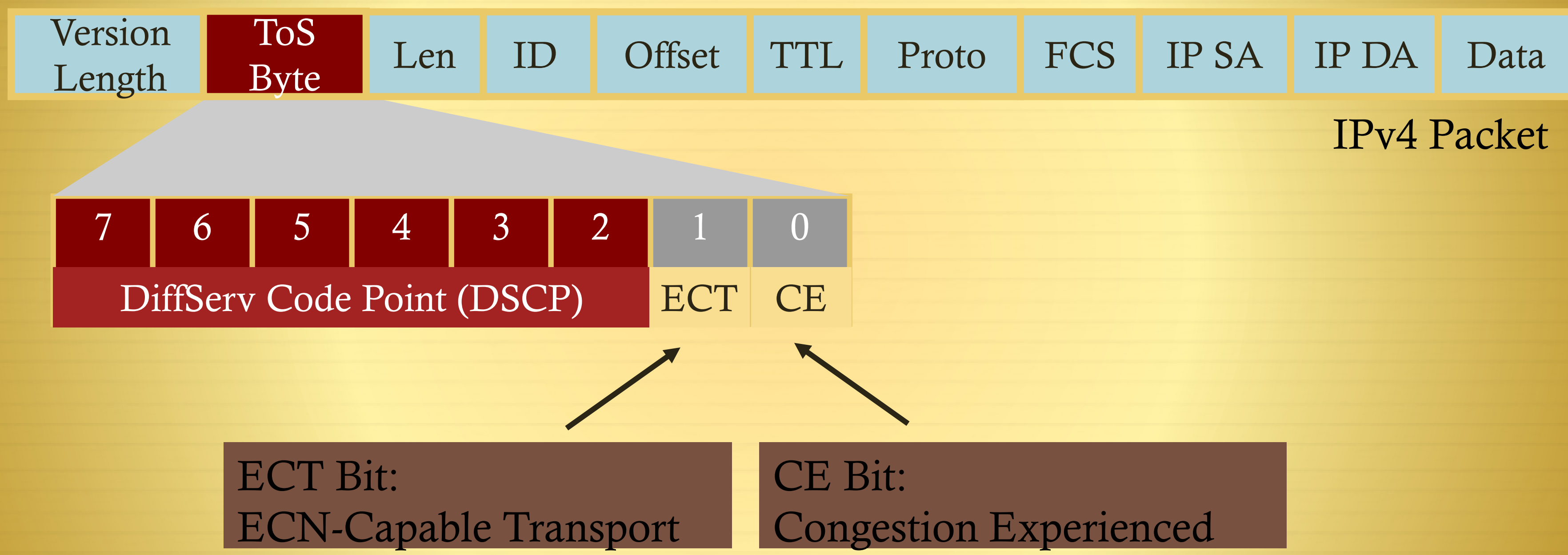
Scheduling Tools
DSCP-Based WRED Operation



AF = (RFC 2597) Assured Forwarding

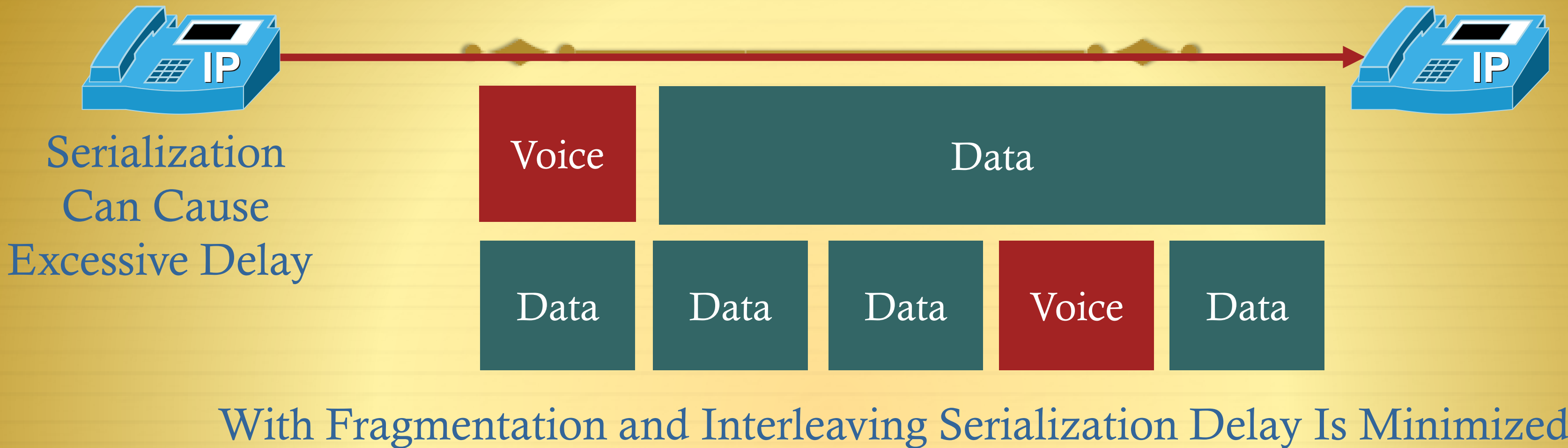
Congestion Avoidance

RFC3168: IP Explicit Congestion Notification



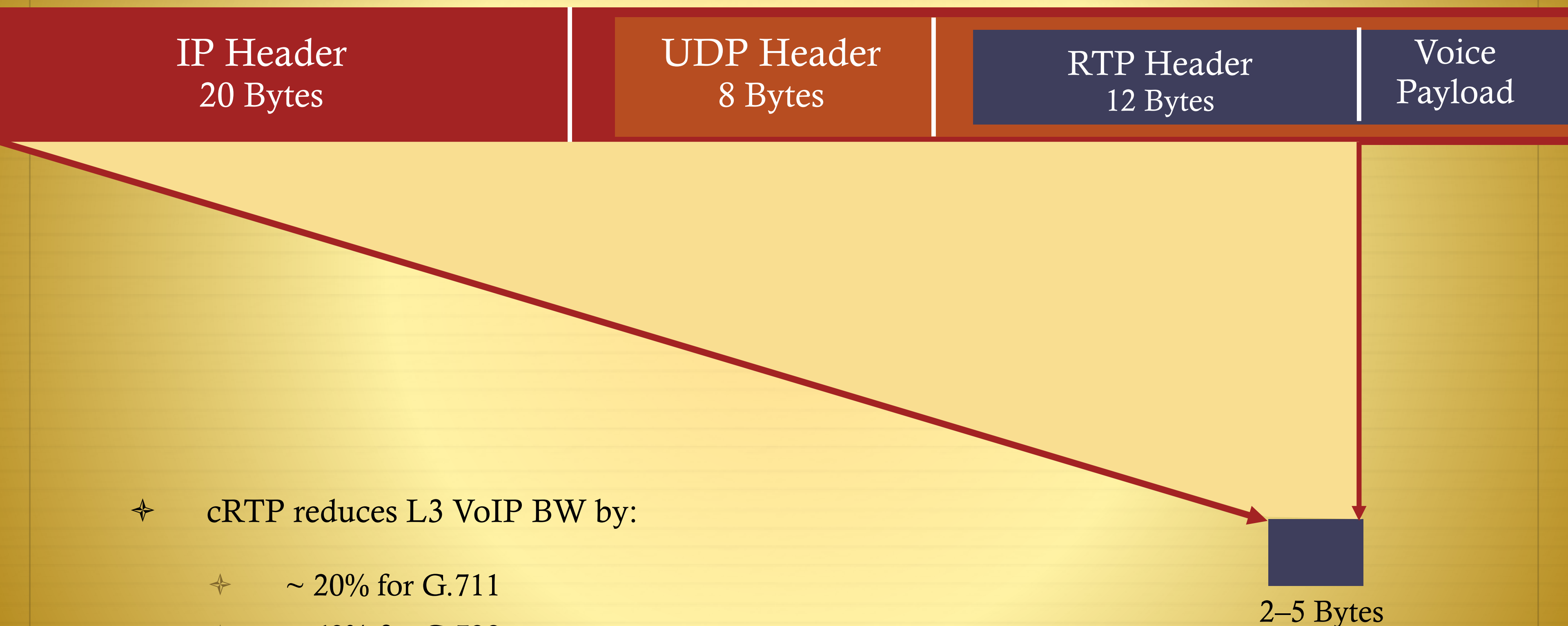
- IP header Type of Service (ToS) byte
- Explicit Congestion Notification (ECN) bits

Link-Specific Tools
Link-Fragmentation and Interleaving



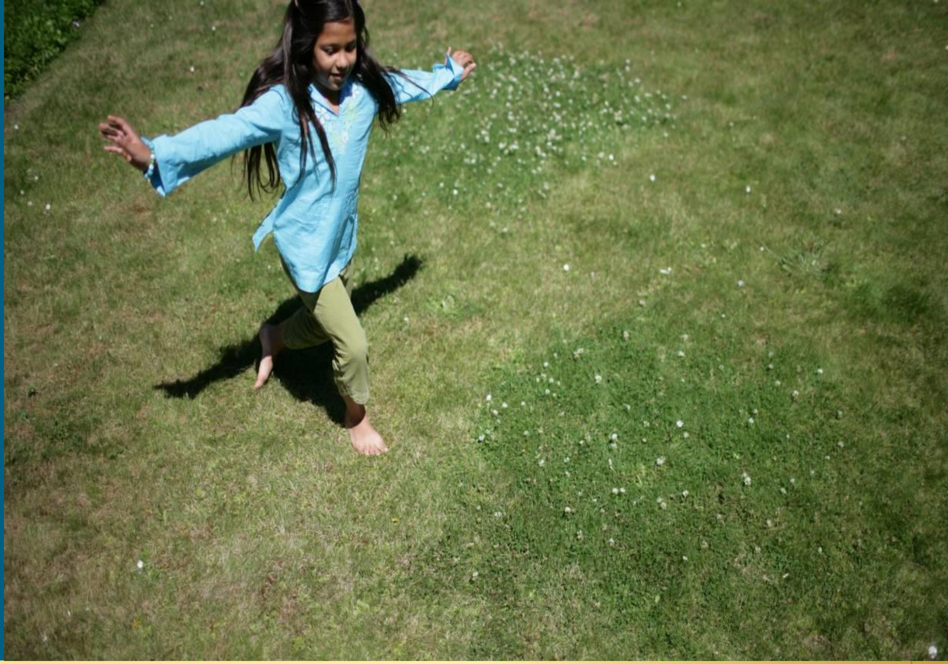
- Serialization delay is the finite amount of time required to put frames on a wire
- For links ≤ 768 kbps serialization delay is a major factor affecting latency and jitter
- For such slow links, large data packets need to be fragmented and interleaved with smaller, more urgent voice packets. Implementation examples: MLPPP LFI and FRF (FRF.12)

Link-Specific Tools
IP RTP Header Compression



- cRTP reduces L3 VoIP BW by:
  - ~ 20% for G.711
  - ~ 60% for G.729

IOS QOS Implementation



What is MQC

- MQC stands for Modular QoS CLI
- Implements the DiffServ model
- Basically: this is how you should configure Quality of Service on Cisco Routers.

```
class-map match-all one
 match ip precedence 5
 match  dscp default
class-map match-all two
 match any
 match  dscp 1
class-map match-all three
 match protocol gnutella
!
policy-map test
 class one
  priority 100
 class two
  bandwidth 300
 class three
   drop
 class class-default
   police 75000 5000
   fair-queue
!
interface Ethernet0/0
 ip address 10.48.77.104 255.255.255.0
 service-policy output test
```

Why was MQC developed ?

- Provide a platform-independent CLI for configuring QoS on Cisco platforms (<>HQF)

- Use standard commands to define a QoS function or a general behavior.

  Defines the syntax and semantics

- Move burden of complexity away from customers, who see functional innovation.

  Hides differences in algorithms or hardware implementation

  No platform specific commands

What is HQF ?

Hierarchical Queuing Framework is a general and scalable infrastructure for supporting a set of QoS features – shaping, low latency queuing, guaranteed bandwidth, flow-based fair queuing, WRED.

To provide support for multiple levels in the queuing hierarchy

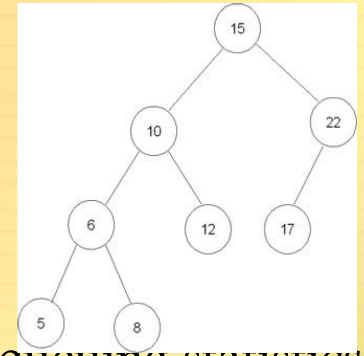
- Translation from user configuration to packet scheduling parameters:

  Minimum guarantee

  Maximum rate

  Excess sharing ratio

  Priority level



- Consistent gathering and displaying of queuing statistics
- Clean separation between control and data plane
- Consistent semantics for queuing features

Configuring QOS using MQC: 3 Steps

1. **Class-map** – To define traffic classes (global config).

2. **Policy-map** – To associate policies/actions with each class of traffic (global config).

3. **Service-policy** – To attach policies to interfaces (logical or physical), in input or output direction (inteface config).

MQC: Step 1 – Class-map

- Creates a named traffic class
- Specifies the packet-matching criteria need to be part of the class.

```
class-map <match-(all|any)> <class name>
  match <criteria>
  match not <criteria>
```

- If more than one criteria, class-map can be ‘match-all’ or ‘match- any’. Default is match all.
- A class named ‘class-default’ is always present, It matches packets that didn’t match a user-defined class.

MQC: Step 2 – Policy-map

- Named object representing a set of policies that are to be applied to a set of traffic classes:

  Ex: Minimum bandwidth guaranteed, maximum rate,…

```
policy-map <map-name>
 class <class-map-name-1>
  <policy-1>
  <policy-n>
  class <class-map-name-n>
  <policy-n>
  class class-default
  <policy-default>
```

- Classes need to be defined first (except class-default)

MQC: Step 3 – Service-policy

- Attach the previously created policy-map to an interface
- Apply it to either input or output traffic

```
service-policy <output|input> <policy-name>
```

- Interface can be physical :

  Main interface

- Or logical :

  Subinterface, PVC, DLCI, Tunnel, Virtual-Template, Dialer, Multilink.

MQC: Hierarchical Policies

- One policy-map can be used inside another one. The parent is the one applied to the interface.

```
policy-map child
     class http
       bandwidth <BW>
     class ftp

policy-map parent
     class class-default
       shape average <CIR>
       service-policy child
```

- Availability and number of levels depends heavily on platform.
- Often used with two levels: Shaper in parent, Queues in child, so the shaper can trigger the backpressure.

Queue Hierarchy

Tree structures made of nodes, leaves and root.

To define how packets will be scheduled.

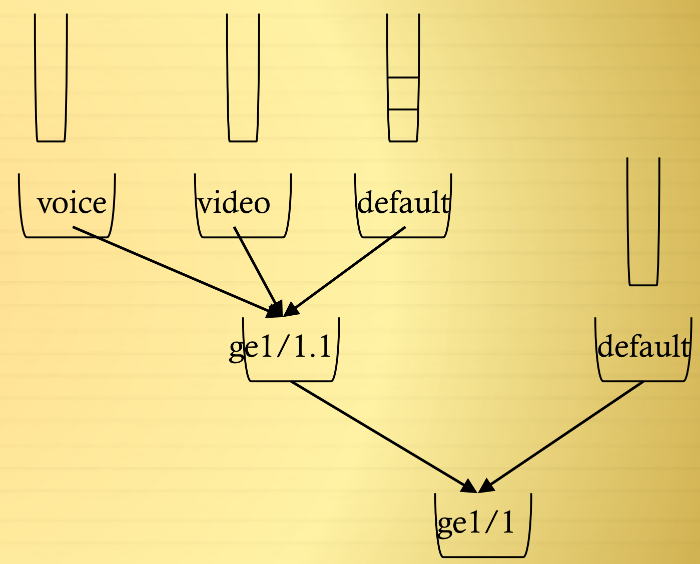
- Root is where the final bottleneck occurs. Most of the time this is the physical interface.
- Classification of a packet will map to a leaf queue in the hierarchy.
- The node defines the scheduling parameters. Three parameters are used: Min BW, Max BW, Excess BW.
- Every level in the HQF hierarchy always has a default queue that captures un-classified traffic at that level

Queue Hierarchy Example

- MQC:

```
Policy-map child
    class voice
        priority level 1 100 kbps
    class video
        bandwidth 2000 kbps
    class class-default
Policy-map parent
    class class-default
        shape average 4000000 bps
    service-policy child

Interface ge1/1.1
    service-policy output parent
```

Hierarchy:



Classification of voice traffic maps to the voice queue

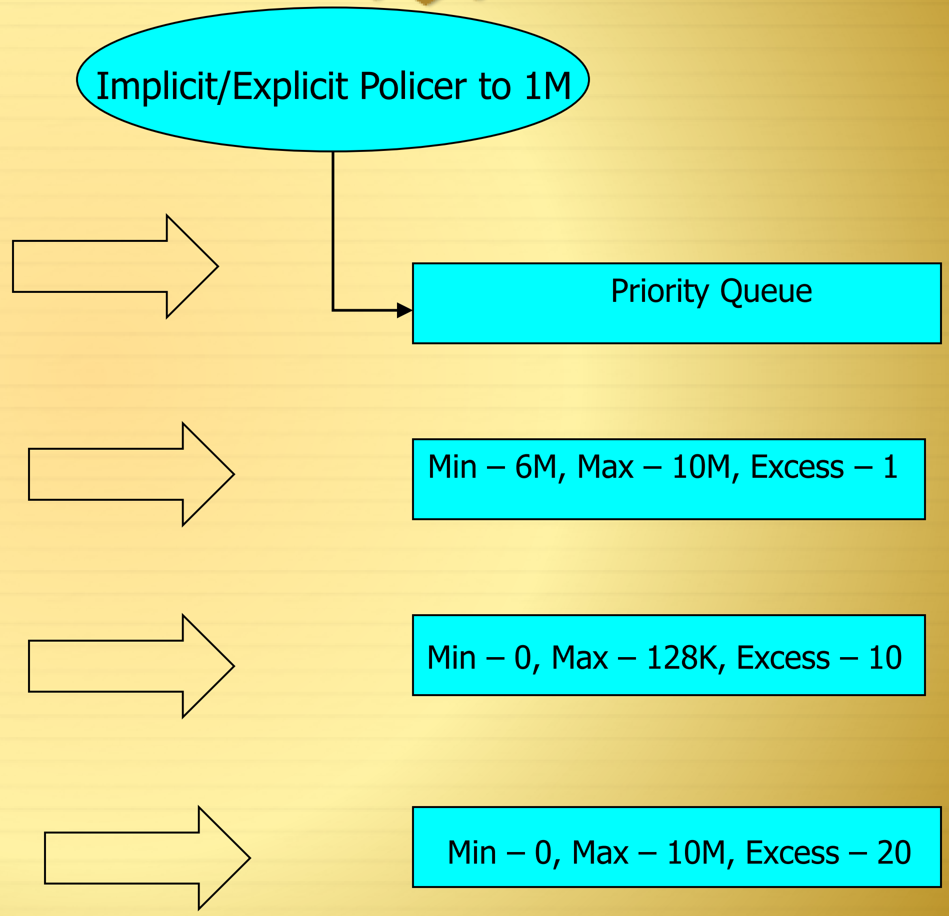
Classification of class-default traffic maps to the default queue that is sibling of voice and video queues

ge1/1 traffic from sub-interfaces other than ge1/1.1 maps to the default queue that is a sibling of the

Queue Hierarchy Example (3 parameter capability)

Assume 10 M interface:

```
policy-map cbwfq
  class voice
    priority percent 10
  class data
    bandwidth percent 60
  class ftp
    bandwidth remaining ratio 10
    shape average 128000
  class class-default
    bandwidth remaining ratio 20
    random-detect
```



HQF: MQC commands

- LLQ

  Priority <kbps>/percent/level

  Conditional/Unconditional Traffic policing (police command)

- Bandwidth

  - *Bandwidth <kbps>/percent/remaining percent/remaining ratio*
  - *<kbps> :* class is guaranteed a minimum allocation of *kbps* kbps
  - *percent :* class is guaranteed x% of the underlying link rate

Note: The **bandwidth** and **priority** commands provide bandwidth guarantees that are often described as bandwidth that is reserved or set aside. However, neither command implements a true reservation of bandwidth. If a traffic class is not using its configured bandwidth, the unused bandwidth is shared among the other classes.

  - *remaining percent :* the bandwidth remaining percent command is used to allocate class 20%of the total remaining (i.e., excess) bandwidth, where total remaining bandwidth is defined as bandwidth not allocated as minimum guarantees to other classes.
  - *remaining ratio:* This number (ratio) indicates the proportional relationship between the class queues. During congestion, the router uses this bandwidth-remaining ratio to determine the amount of excess bandwidth to allocate to a class of nonpriority traffic

HQF: Supported MQC features

- Police
  - Single Rate Three Color Marker implementation:
  - *police cir <bps>/percent <%> bc <bc> be <be> conform <conform-action> exceed <exceed-action> violate <violate-action>*
  - Two Rate Three Color Marker implementation:
  - *police cir <bps> bc <bc> pir <pir> be <be> conform <conform-action> exceed <exceed-action> violate <violate-action>*
- Shape

  Shape average/peak <bps>/percent <value> <bc> ms <be> ms

  - The 'shape peak ...‘ version of the command is targeted at frame-relay environments where the frame relay network accepts bc + be bits per interval, but may mark the excess traffic with the discard eligible (DE) bit.Thus it is desirable for a router to have the capability to send bc + be bits per interval when connected to a frame-relay cloud that allows/expects this behavior.

HQF: Supported MQC features

- Fair-Queue – Flow based!
  - The fair-queue command provides fair bandwidth allocation among IP "flows" within a class of traffic. The flows are defined by a hash on the 5-tuple (source address, destination address, source port, destination port, protocol). The fair-queue action provides for fair access to bandwidth among flows within a class (i.e,. each flow gets an equal share of the bandwidth), as well as fair access to buffers among flows within a class (i.e., each flow gets an equal share of the buffers)

  fair-queue [queue-limit <individual-limit>]

- WRED
  - *The random-detect command is used to enable [W]RED on a class of traffic. Drop-probability controls the probability of dropping the packet when the queue size reaches the maximum threshold*

  Random-detect precedence/dscp/cos/clp min-threshold <value> bytes/packets/ms max-threshold <value> bytes/packets/ms drop-probability <value>

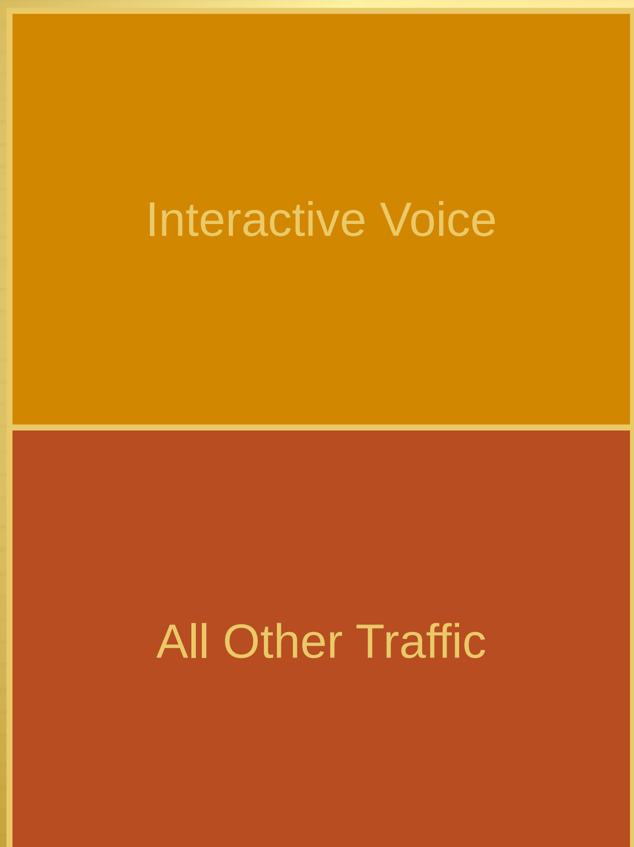
- Queue-limit
  - The queue-limit command is used to tune the limit on the queue associated with a particular class of traffic. The command takes one parameter, which defines the maximum depth the queue is allowed to reach prior to tail drop occurring. The depth of the queue can be specified in units of packets, bytes/kbytes/mbytes/gbytes, or in terms of the time it takes to drain the queue at its minimum guaranteed service rate.

  queue-limit <value> packets/bytes/ms

Cisco AutoQoS:
Two Offerings, Two Levels of Detail

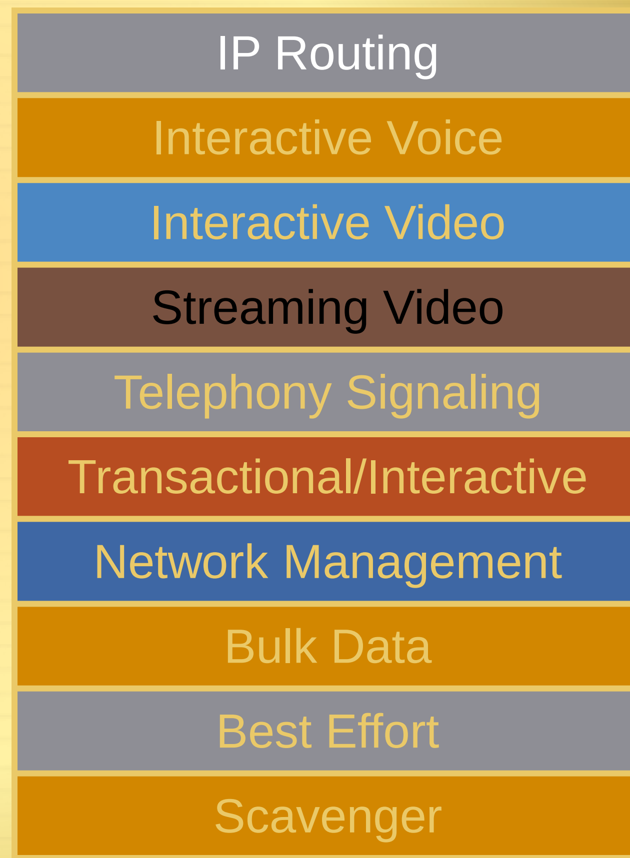
AutoQoS—VoIP

Focus on Voice vs. Data



AutoQoS—Enterprise

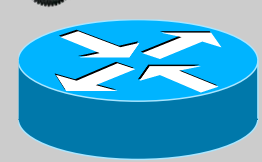
Up to 10 Classes



AutoQoS
AutoQoS VoIP: WAN

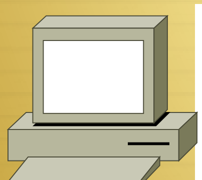
```
interface Serial2/0
 bandwidth 768
 ip address 10.1.102.2 255.255.255.0
 encapsulation ppp
 auto qos voip trust
!
 class-map match-any AutoQoS-VoIP-RTP-Trust
  match ip dscp ef
 class-map match-any AutoQoS-VoIP-Control-Trust
  match ip dscp cs3
  match ip dscp af31
!
!
 policy-map AutoQoS-Policy-Trust
  class AutoQoS-VoIP-RTP-Trust
   priority percent 70
  class AutoQoS-VoIP-Control-Trust
   bandwidth percent 5
  class class-default
   fair-queue
!
```

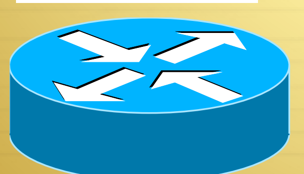
```
!
interface Multilink2001100117
 bandwidth 768
 ip address 10.1.102.2 255.255.255.0
 service-policy output AutoQoS-Policy-Trust
 ip tcp header-compression iphc-format
 no cdp enable
 ppp multilink
 ppp multilink fragment delay 10
 ppp multilink interleave
 ppp multilink group 2001100117
 ip rtp header-compression iphc-format
!
…
!
interface Serial2/0
 bandwidth 768
 no ip address
 encapsulation ppp
 auto qos voip trust
 no fair-queue
 ppp multilink
 ppp multilink group 2001100117
!
```



AutoQoS
AutoQoS Enterprise: WAN DiffServ Classes

AutoDiscovery	Cisco AutoQoS Policy
Application and Protocol Types	Cisco AutoQoS Class-Maps Match Statements
Offered Bit Rate (Average and Peak)	Minimum Bandwidth to Class Queues, Scheduling and WRED



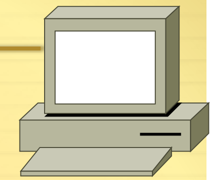



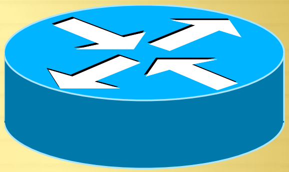
Traffic Class	DSCP
IP Routing	CS6
Interactive Voice	EF
Interactive Video	AF41
Streaming Video	CS4
Telephony Signaling	CS3
Transactional/Interactive	AF21
Network Management	CS2
Bulk Data	AF11
Best Effort	0
Scavenger	CS1

AutoQoS

AutoQoS Enterprise: WAN, Part One: Discovery

AutoDiscovery Notes

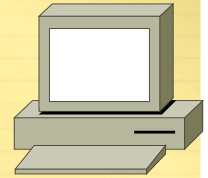


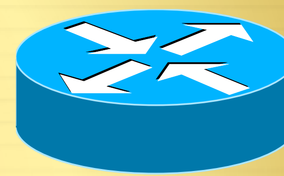



```
interface Serial4/0 point-to-point
encapsulation frame-relay
bandwidth 256
ip address 10.1.71.1 255.255.255.0
frame-relay interface-dlci 100
  auto discovery qos
```

- Command should be enabled on interface of interest
- Do not change interface bandwidth when running auto discovery
- Cisco Express Forwarding must be enabled
- All previously attached QoS policies must be removed from the interface

AutoQoS Enterprise: WAN, Part One: Discovery (Cont.)



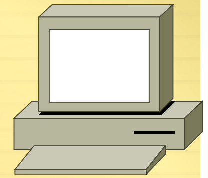
```
Router# show auto discovery qos

AutoQoS Discovery enabled for applications
 Discovery up time: 2 days, 55 minutes
 AutoQoS Class information:
 Class VoIP:
  Recommended Minimum Bandwidth: 517 Kbps/50% (PeakRate)
  Detected applications and data:
  Application/         AverageRate          PeakRate      Total
  Protocol             (kbps/%)             (kbps/%)      (bytes)
  rtp audio            76/7                 517/50        703104
 Class Interactive Video:
  Recommended Minimum Bandwidth: 24 Kbps/2% (AverageRate)
  Detected applications and data:
  Application/         AverageRate          PeakRate       Total
  Protocol             (kbps/%)             (kbps/%)       (bytes)
  rtp video            24/2                 5337/52        704574
 Class Transactional:
  Recommended Minimum Bandwidth: 0 Kbps/0% (AverageRate)
  Detected applications and data:
  Application/         AverageRate          PeakRate       Total
  Protocol             (kbps/%)             (kbps/%)       (bytes)
  citrix                 36/3                  74/7        30212
  sqlnet                 12/1                  7/<1        1540
```

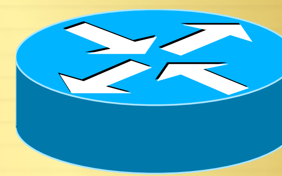
AutoQoS Enterprise: WAN, Part Two: Provisioning

```
interface Serial4/0 point-to-point
 bandwidth 256
 ip address 10.1.71.1 255.255.255.0
 frame-relay interface-dlci 100
  auto qos
   class-map match-any AutoQoS-Voice-Se4/0
    match protocol rtp audio
   class-map match-any AutoQoS-Inter-Video-Se4/0
    match protocol rtp video
   class-map match-any AutoQoS-Transactional-Se4/0
    match protocol sqlnet
    match protocol citrix
  !
  policy-map AutoQoS-Policy-Se4/0
     class AutoQoS-Voice-Se4/0
      priority percent 70
      set dscp ef
     class AutoQoS-Inter-Video-Se4/0
      bandwidth remaining percent 10
      set dscp af41
     class AutoQoS-Transactional-Se4/0
      bandwidth remaining percent 1
      set dscp af21
     class class-default
      fair-queue
  !
```

AutoQoS Enterprise: WAN, Part Two: Provisioning (Cont.)

```
interface Serial4/0 point-to-point
 bandwidth 256
 ip address 10.1.71.1 255.255.255.0
 frame-relay interface-dlci 100
  auto qos
```

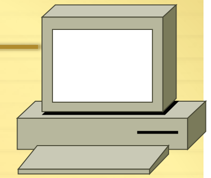


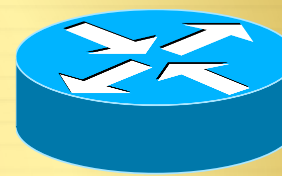



```
<policy continued>
!
policy-map AutoQoS-Policy-Se4/0-Parent
   class class-default
   shape average 256000
    service-policy AutoQoS-Policy-Se4/0
!
interface Serial4/0 point-to-point
  frame-relay interface-dlci 100
   class AutoQoS-FR-Serial4/0-100
!
map-class frame-relay AutoQoS-FR-Serial4/0-100
frame-relay cir 256000
frame-relay mincir 256000
frame-relay fragment 320
service-policy output AutoQoS-Policy-Se4/0-Parent
```

AutoQoS Enterprise: WAN, Part Three: Monitoring

toring Drops in LLQ



- Thresholds are activated in RMON alarm table to monitor drops in Voice Class
- Default drop threshold is 1bps

```
rmon event 33333 log trap AutoQoS description "AutoQoS
SNMP traps for Voice Drops" owner AutoQoS

rmon alarm 33350 cbQoSCMDDropBitRate.2881.2991 30
Absolute rising-threshold 1 33333 falling-threshold 0
Owner AutoQoS
```

RMON Event Configured and Generated by Cisco AutoQoS

QoS Best-Practice Design Principles

Classification and Marking Design
Where and How Should Marking Be Done?

- QoS policies (in general) should always be performed in hardware, rather than software, whenever a choice exists
- Classify and mark applications as close to their sources as technically and administratively feasible
- Use DSCP markings whenever possible
- Follow standards-based DSCP PHBs to ensure interoperation and future expansion
  - RFC 2474 Class Selector Code Points
  - RFC 2597 Assured Forwarding Classes
  - RFC 3246 Expedited Forwarding

Classification and Marking Design
QoS Baseline Marking Recommendations

Application	L3 Classification			L2
	IPP	PHB	DSCP	CoS
Routing	6	CS6	48	6
Voice	5	EF	46	5
Video Conferencing	4	AF41	34	4
Streaming Video	4	CS4	32	4
Mission-Critical Data	3	AF31*	26	3
Call Signaling	3	CS3*	24	3
Transactional Data	2	AF21	18	2
Network Management	2	CS2	16	2
Bulk Data	1	AF11	10	1
Best Effort	0	0	0	0
Scavenger	1	CS1	8	1

Policing Design Principles
Where and How Should Policing Be Done?

- Police traffic flows as close to their sources as possible
- Perform markdown according to standards-based rules, whenever supported
  - RFC 2597 specifies how assured forwarding traffic classes should be marked down (AF11 → AF12 → AF13) which should be done whenever DSCP-based WRED is supported on egress queues
  - Cisco Catalyst platforms currently do not support DSCP-based WRED, so Scavenger-class remarking is a viable alternative
  - Additionally, non-AF classes do not have a standards-based markdown scheme, so Scavenger-class remarking is a viable option

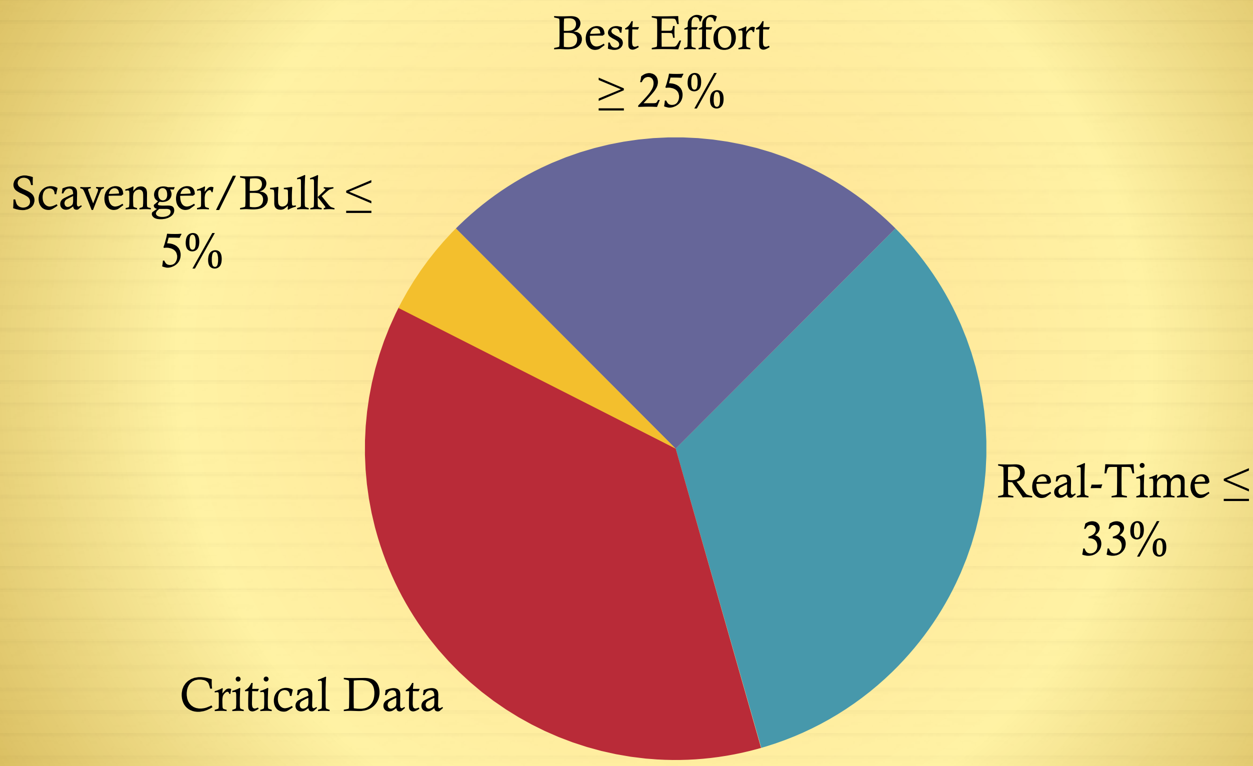
Queuing Design Principles

Where and How Should Queuing Be Done?

- The only way to provide service guarantees is to enable queuing at any node that has the potential for congestion
  - Regardless of how rarely—in fact—this may occur
- At least 25 percent of a link's bandwidth should be reserved for the default Best Effort class
- Limit the amount of strict-priority queuing to 33 percent of a link's capacity
- Whenever a Scavenger queuing class is enabled, it should be assigned a minimal amount of bandwidth
- To ensure consistent PHBs, configure consistent queuing policies in the Campus + WAN + VPN, according to platform capabilities
- Enable WRED on all TCP flows, whenever supported
  - Preferably DSCP-based WRED

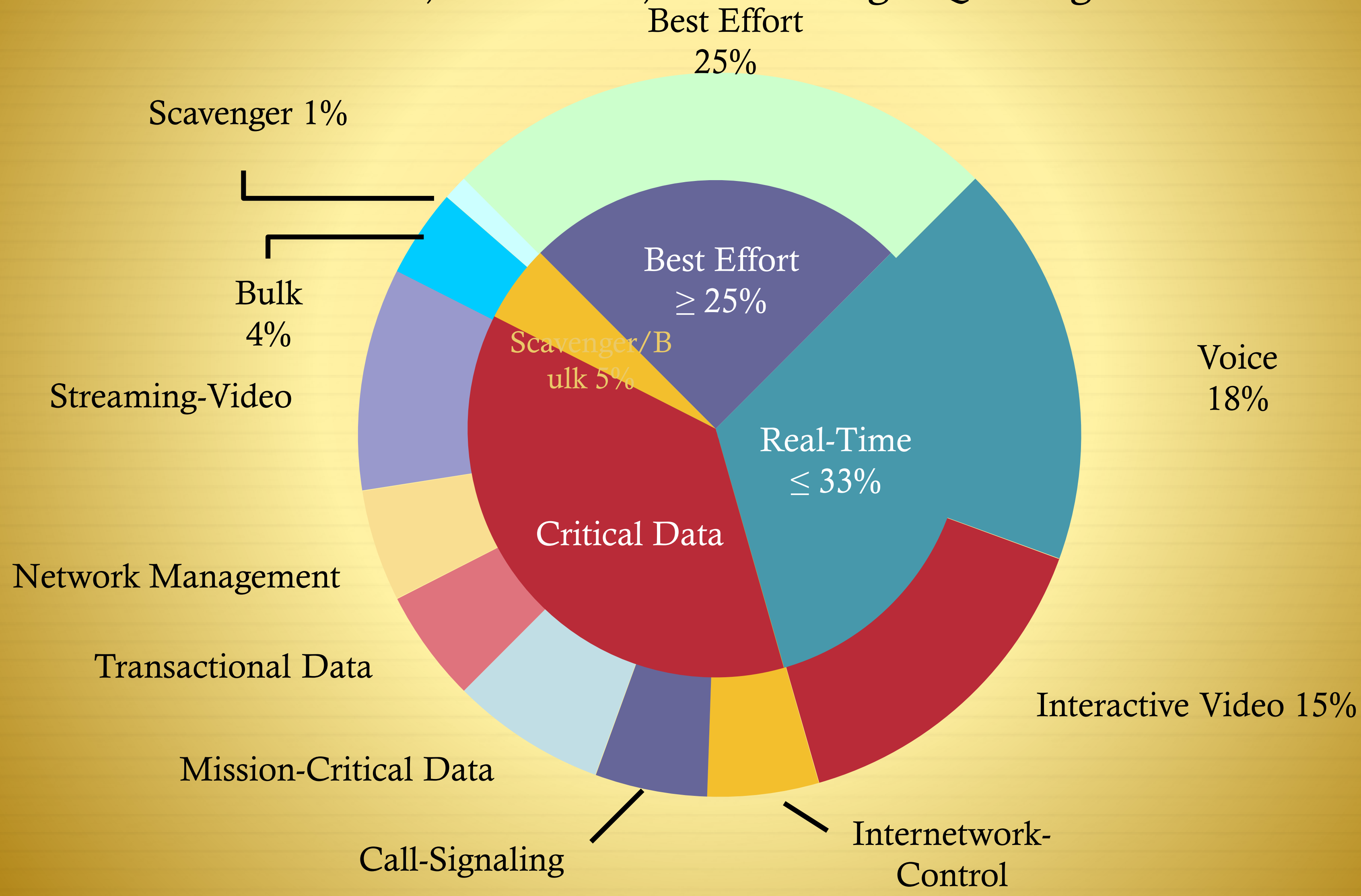
Campus Queuing Design

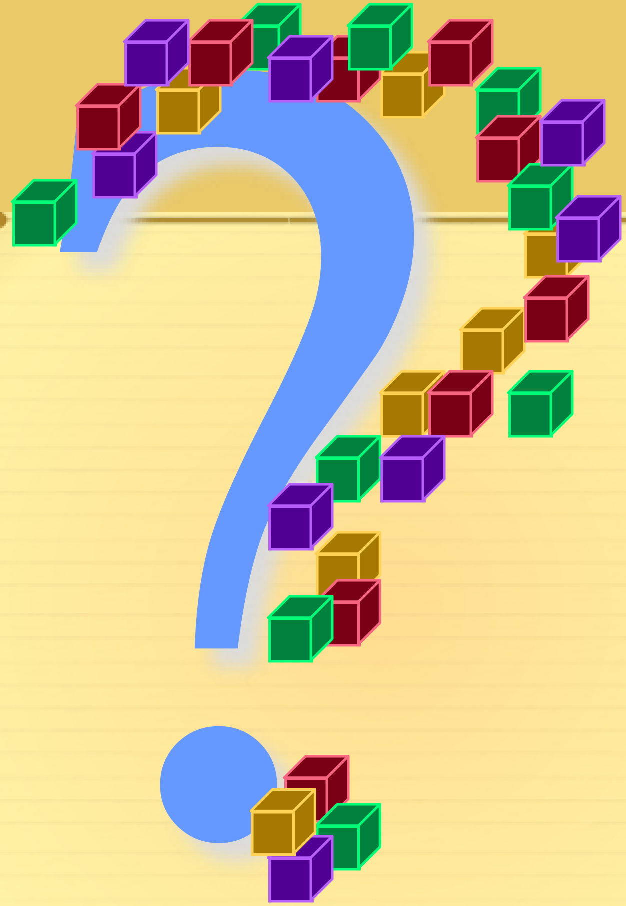
Realtime, Best Effort, and Scavenger Queuing Rules



Campus and WAN/VPN Queuing Design

Compatible Four-Class and Eleven-Class Queuing Models Following Realtime, Best Effort, and Scavenger Queuing Rules





Questions???