Quality of Service in the Internet

Theory and Practice

Geoff Huston

Acknowledgment and thanks

to Fred Baker and Paul Ferguson, both of Cisco Systems,

for some of the material used in this slide set

Agenda

- Definition of the Problem
- Service Quality and the Application Environment
- Approaches to Quality Management
- Considerations
- Mechanisms
- Measuring QoS
- Marketing QoS
- Summary

Quality of Service

Definition of the Problem

What is the problem?

- Today’s Internet is plagued by sporadic poor performance.

This is getting worse, not better!

- Methods are needed to differentiate traffic and provide “services”

What is the problem?

- “Poor Performance” of the Internet service environment
  - more specifically:
    - routing instability
    - high packet loss in critical NAPs/Exchange Points
    - server congestion
    - high variation of transaction times
    - poor protocol performance due to loss and round trip time variation

The QoS challenge

- Internet network infrastructure is under stress due to:
  - robust demand models
  - engineering the network to use all available resources, on the edge of instability and capacity saturation

What is the problem?

- Applications are more demanding
  - end systems are getting faster
  - end systems use faster network connections
  - emerging ubiquity of access breeds diversity of application requirements
  - end systems applications wish to negotiate performance from the network

It would be good if…

- When the user wished to access a priority service:
  - the network could honor the request
- The application could forecast its network load requirements so that:
  - the network could commit to meet them
- When there isn't sufficient bandwidth on one network path to meet the application’s requirements:
  - The network could find another path

Quality of Service

- Customers want access to an Internet service which can provide a range of **consistent & predictable high quality service levels**
  - in addition to normal best effort service levels

Quality of Service

- Network mechanisms intended to meet this demand for various levels of service are categorized within the broad domain of **Quality of Service**

Rationale for providing QoS

- The Internet is commercial & competitive.
  - No major revelation here
- Internet Service Providers are looking for ways to generate new sources of revenue.
  - Again, nothing new
- Creation of new services creates new sources of revenue.

Rationale for providing QoS

- Preferential treatment is an attractive service which customers are indicating they desire to purchase.

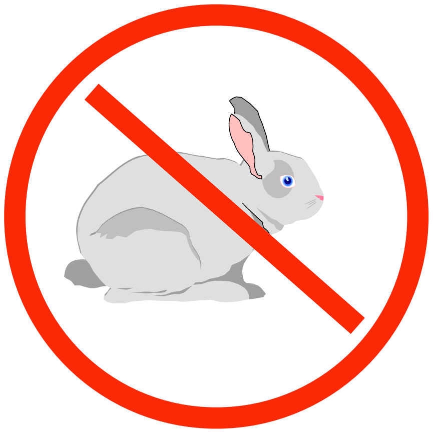
Rationale for providing QoS

- Service Providers would like offer differential services where:
  - the customer is charged at a rate comparable to service level expectations
  - where the marginal service revenue reflects the marginal network engineering and support costs for the service

Non-Rationale for QoS

- QoS is not a tool to compensate for inadequacies elsewhere in the network.
- It will not fix:
  - Massive over-subscription
  - Horrible congestion situations
  - Sloppy network design

No magic here™



What is the expectation?

- A requirement for a ***premium differentiated*** services within the network that will provide ***predictable*** and ***consistent*** service response for ***selected customers*** and traffic flows:
  - provision of **guarantees** on bandwidth & delay
  - provision of **absolute** service level agreements
  - provision of **average** service level agreements

But can the Internet deliver?

QoS

- QoS is the provision of control mechanisms within the network which are intended to manage congestion events.

Aside...

- The Congestion Problem
  as we see it --
  - Chaos theory:
    - Congestion is non-linear behavior
    - Think in terms of pipes and water
    - Turbulence produces mixing and increases drag
  - QoS is akin to solving non-linear fluid dynamics:
    - Enforcing linearity, or
    - Convincing big flows to behave nicely

Expectation setting

QoS is not magic

- QoS cannot offer cures for a poorly performing network
- QoS does not create nonexistent bandwidth.
  - Elevating the amount of resources available to one class of traffic decreases the amount available for other traffic classes
  - Total goodput will be reduced in a differentiated environment
- QoS will not alter the speed of light
  - On an unloaded network, QoS mechanisms will not make the network any faster
    - Indeed, it could make it slightly worse!

Expectation setting

QoS is unfair damage control

- QoS mechanisms attempt to preferentially allocate resources to predetermined classes of traffic, when the resource itself is under contention
- The preferential allocation can be wasteful, making the cumulative damage worse
- Resource management only comes into play when the resource is under contention by multiple customers or traffic flows
  - Resource management is irrelevant when the resource is idle or not an object of contention

Expectation setting

QoS is relative, not absolute

- QoS actively discriminates between preferred and non-preferred classes of traffic at those times when the network is under load (congested)
- Qos is the relative difference in service quality between the two generic traffic classes
  - If every client used QoS, then the net result is a zero sum gain

Expectation setting

QoS is intentionally elitist and unfair

- The QoS relative difference will be greatest when the preferred traffic class is a small volume compared to the non-preferred class
- QoS preferential services will probably be offered at a considerable price premium to ensure that quality differentiation is highly visible

Quality of Service

Definition of *Quality*

What is *Quality*?

- *Quality* cannot be measured on an entire network
  - Flow bandwidth is dependant on the chosen transit path
  - Congestion conditions are a localized event
  - Quality metrics degrade for those flows which transit the congested location
- *Quality* can be only be measured on an end-to-end traffic flow, at a particular time

Quality **metrics**

- The network quality metrics for a flow are:
  - ***Delay*** - the elapsed time for a packet to transit the network
  - ***Jitter*** - the variation in delay for each packet
  - ***Bandwidth*** - the maximal data rate that is available for the flow
  - ***Reliability*** - the rate of packet loss, corruption, and re-ordering within the flow

Quality metrics

- Quality metrics are amplified by network load
  - ***Delay*** increases due to increased queue holding times
  - ***Jitter*** increases due to chaotic load patterns
  - ***Bandwidth*** decreases due to increased competition for access
  - ***Reliability*** decreases due to queue overflow, causing packet loss

QoS and the Internet

- The Internet transmission model is a set of self-adjusting traffic flows that cooperate to efficiently load the network transmission circuits
  - session performance is variable
  - network efficiency is optimised

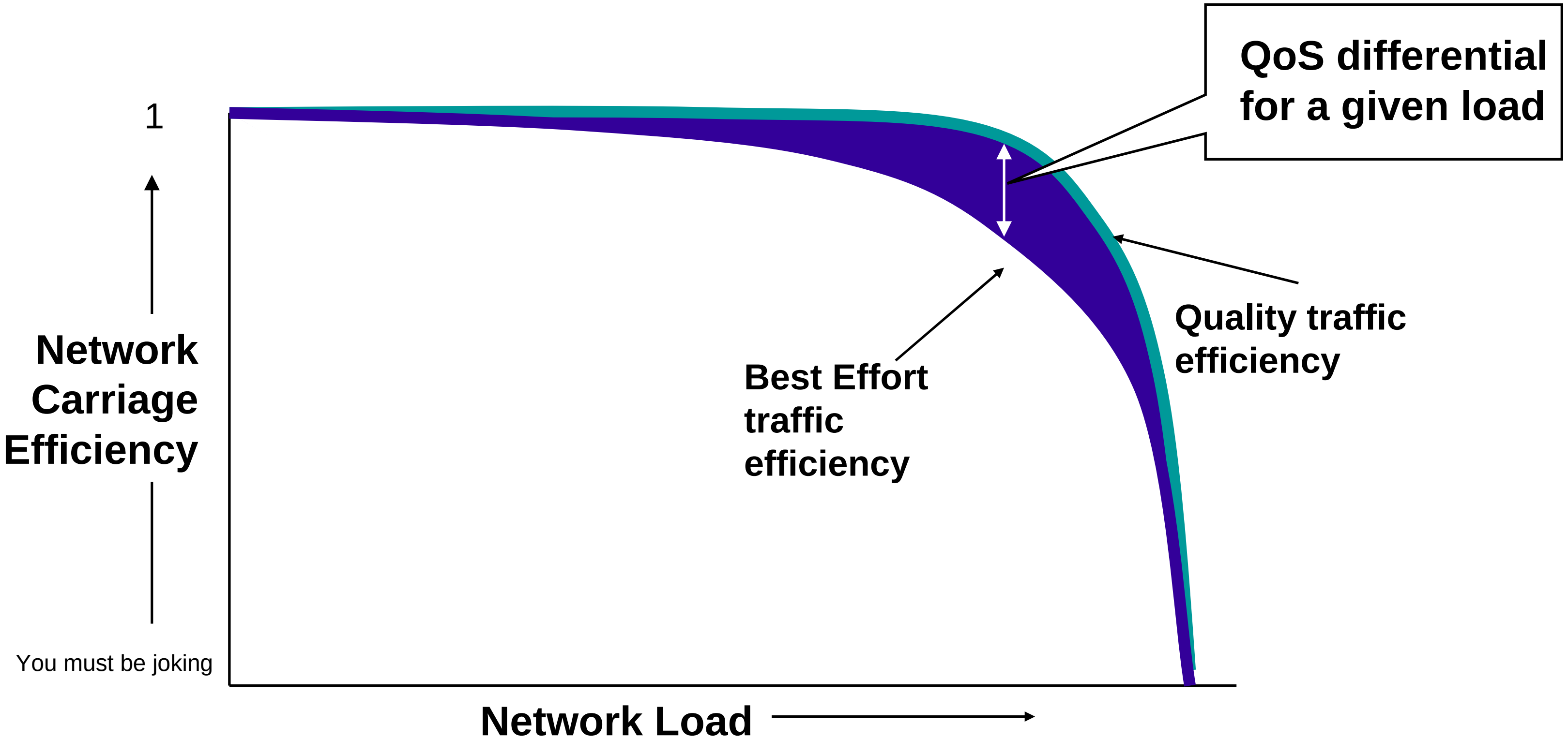
QoS and the Internet

- QoS is a requirement for the network to bias the flow self-adjustment to allow some flows to consume greater levels of the network resource
  - this is not easy...

Quality metrics

- Quality differentiation is only highly visible under heavy network load
  - differentiation is relative to normal best effort
  - On unloaded networks queues are held short, reducing queue holding time, propagation delay is held constant and the network service quality is at peak attainable level

The Internet QoS Margin is small



Service Quality

Not every network is designed with quality in mind…

- Adherence to fundamental networking engineering principals.
- Operate the network to deliver Consistency, Stability, Availability, and Predictability.
- Cutting corners is not necessarily a good idea.

Service Quality

Without Service Quality, QoS is unachievable

Quality of Service

Service Quality and the Application Environment

Application Performance Issues

Today’s Internet Load

- Two IP protocol families
  - TCP
  - UDP
- Three common application elements
  - WWW page fetches (TCP)
  - bulk data transfer (TCP)
  - audio & video transfer (UDP)

consume some 90% of today’s Internets

UDP-based applications

- sender transmits according to external signal source timing, such as:
  - audio encoder
  - video encoder
- one (unicast) or more (multicast) receivers
- no retransmission in response to network loss
    - need to maintain integrity of external clocking of signal
- no rate modification due to network congestion effects
    - no feedback path from network to encoder

UDP and Quality

- QoS: reduce loss, delay and jitter
  - RSVP approach
    - ‘reserve’ resource allocation across intended network path
    - guaranteed load for constant traffic rate encoder
    - controlled load for burst rate managed encoder
  - application-based approach
    - introduce feedback path from receiver(s) to encoder to allow for some rate adjustment within the encoder
  - diff-serv approach
    - mark packets within a flow to trigger weighted preferential treatment

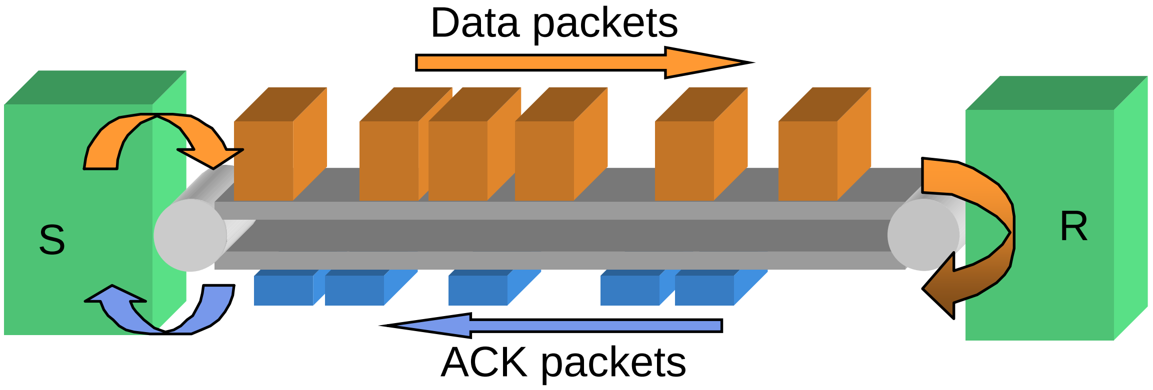
TCP behavior

- Large volume TCP transfers
  - allow the data rate to adjust to the network conditions
  - establish point of network efficiency, then probe it
  - variable rate continually adjusted to optimize network load at the point of maximal transfer without loss
  - uses dynamic adjustment of sending window to vary the amount of data held ‘in flight’ within the network

TCP performance

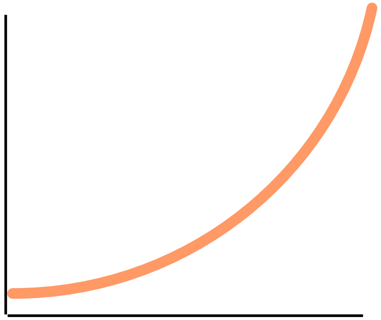
use the Principle of Network Efficiency:

- only inject more data into a loaded network when you believe that the receiver has removed the same amount of data from the network
- TCP uses ACKs as the sender's timer



TCP rate control (1)

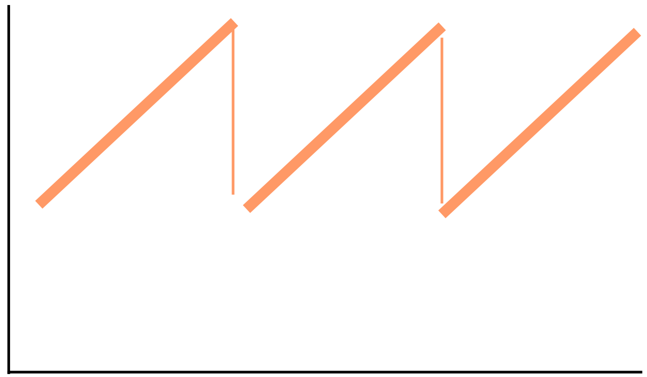
- Slow Start
  - inject one segment into the network, wait for ACK
  - for each ACK received inject ACK'ed data quantity, plus an additional segment (exponential rate growth)
  - continue until fast packet loss, then switch to Congestion Avoidance



Under slow start TCP window growth is exponential

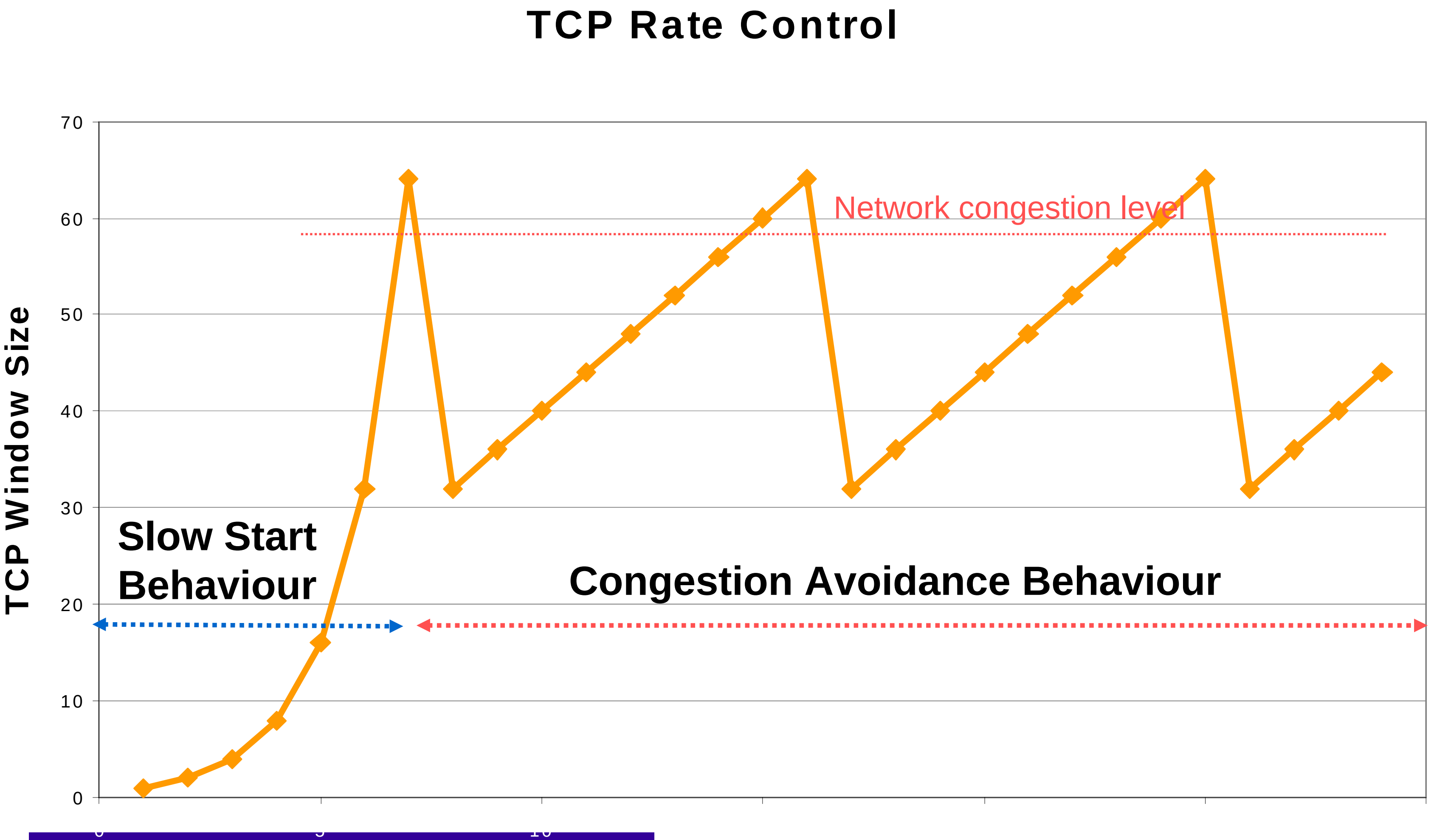
TCP rate control (2)

- Congestion Avoidance
  - halve current window size
  - for each ACK received inject ACK'd data quantity plus message_segment / RTT additional data (linear rate growth of 1 segment per RTT)
  - on fast loss, halve current window size



Under Congestion Avoidance TCP window size is a linear sawtooth

TCP session behaviour



TCP and Quality

- For long held sessions an optimal transfer rate is dependant on:
  - avoiding sequenced packet drop, to allow TCP fast retransmit algorithm to trigger
    - i.e., tail drop is a Bad Thing ™
  - avoiding false network load signals, to allow slow start to reach peak point of path load (ATM folk please take careful note!)
    - congestion avoidance has (slow) linear growth while slow start uses (faster) exponential growth
  - avoiding resonating cyclical queue pressure
    - packets tend to cluster at RTT epoch intervals, needing large queues to even out load at bottleneck spots

Short TCP sessions

- average WWW session is 15 packets
- 15 packets are 4 RTTs under slow start
- average current network load is 70% WWW traffic
- performance management for short TCP sessions is important today

Short TCP and Quality

- Increase initial slow start TCP window from 1 to 4 segments
  - decrease transfer time by 1 RTT for 15 packet flows
- avoid loss for small packet sequences
  - retransmission has proportionately high impact on transfer time
- use T/TCP to avoid 3-way handshake delay
  - reduce transfer time from 6 RTT to 4 RTT
- use HTTP/1.1 to avoid multiple short TCP sessions

TCP and QoS

- TCP performance is based on round trip path
- Partial QoS measures may not improve TCP performance
  - QoS symmetry
  - End-to-end QoS

QoS Symmetry

- Forward (data) precedence without reverse (ACK) precedence may not be enough for TCP
  - data transmission is based on integrity of reverse ACK timing
  - unidirectional QoS setting is not necessarily enough for TCP
    - ACKs should mirror the QoS of the data it acknowledges to ensure optimal performance differential
    - will the network admit such ‘remote setting’ QoS?

      How?
    - will the QoS tariff mechanism support remote triggered QoS?

      How?

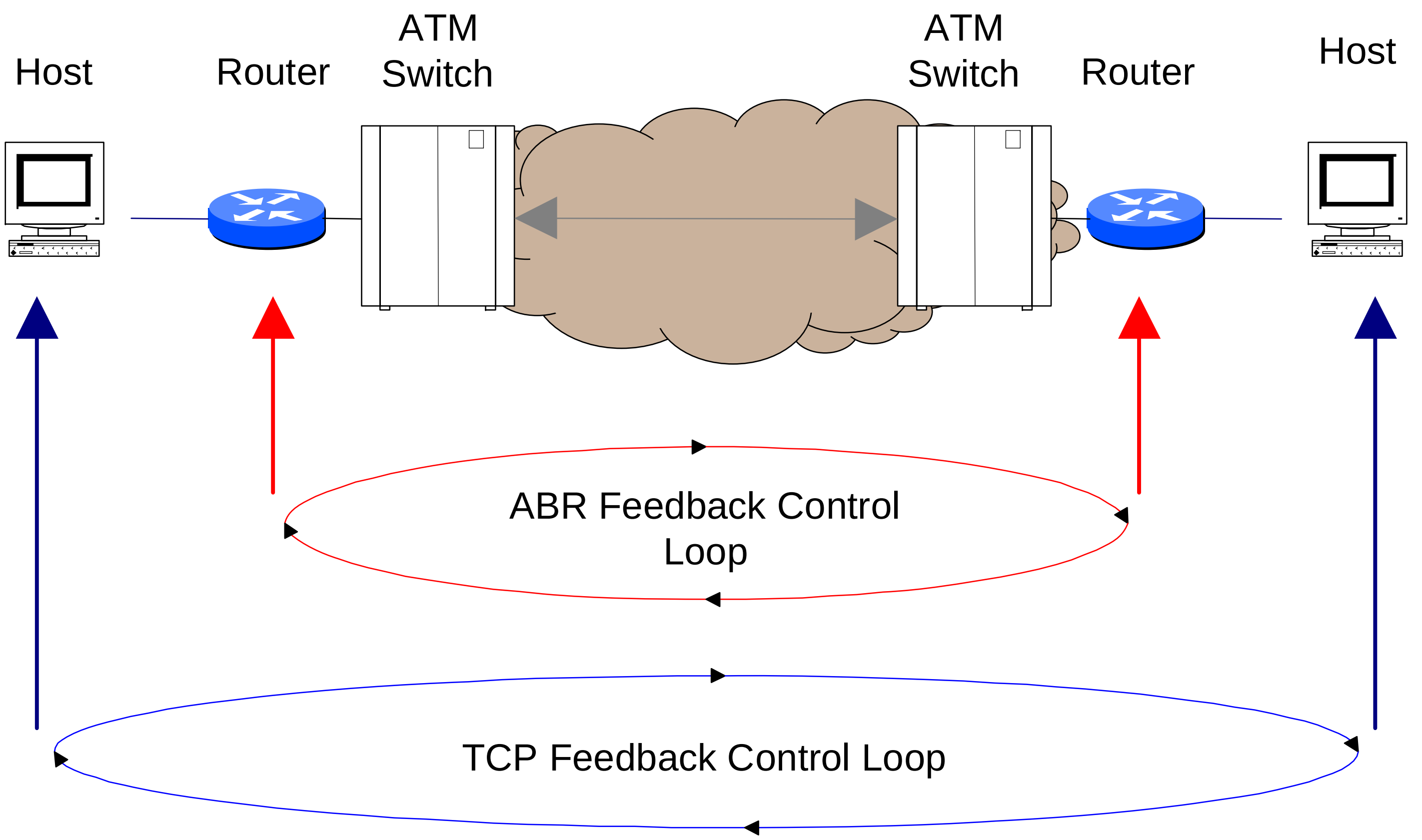
End-to-End QoS

- Precedence on only part of the end-to-end path may not be enough
  - data loss and jitter introduced on non-QoS path component may dominate end-to-end protocol behavior

End-To-End QoS

- Partial provider QoS is not good enough
  - inter-provider QoS agreements an essential precondition for Internet-wide QoS
  - inter-provider QoS agreements must cover uniform semantics of QoS indicators
  - inter-provider agreements are not adequately robust today to encompass QoS

What is End-to-End?



QoS Discovery protocol

- Will we require QoS discovery probes to ‘uncover’ QoS capability on a path to drive around the non-uniform deployment environment?
  - similar to MTU discovery mechanism used to uncover end-to-end MTU
  - use probe mechanism to uncover maximal attainable QoS setting on end-to-end path
    - even if we need it, we haven’t got one of these tools yet!

End-To-End QoS

- BUT --
  - link-based QoS on critical bottleneck paths may produce useful QoS outcomes without complete end-to-end QoS structures
  - Potential hop-based QoS deployment scenarios:
    - queuing precedence on heavily congested high delay link
      - place mechanism on critical common bottleneck point
    - satellite vs cable QoS path selection
      - use policy-based forwarding for path selection

Quality of Service

Service Quality and the Application Environment

Delay Management

Operational definition

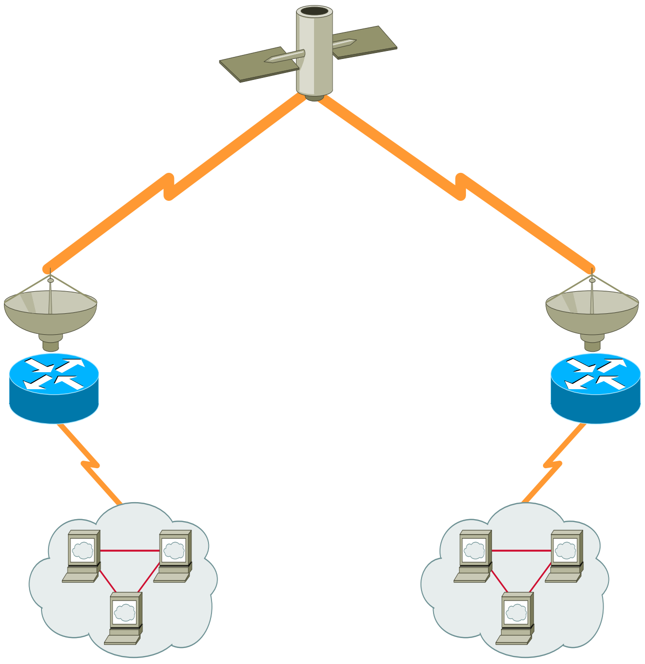
- Application perspective:
  - A link or network over which an application is less useful due to the effects of delay
- User Perspective:
  - the World Wide Wait

TCP measures delay

- Mean Round Trip Time (RTT)
  - elapsed time for a data packet to be sent and a corresponding ACK packet to be received
- One window of data per RTT
- Mean variance in RTT
- Retransmit after
  - Mean RTT + (constant x Mean Variance)

Delay affects performance

What is delay?



- Packet propagation times
  - LAN - less than 1 millisecond
  - campus - 1 millisecond
  - trans-US - 12 milliseconds
  - trans-Pacific cable - 60 milliseconds
  - AU to US - 120 milliseconds
  - AU to FI - 220 milliseconds
  - LEO - variable - 100 - 200 milliseconds
  - GEO - 280 milliseconds

Delay and applications

- One window transaction per RTT
- Small transmission windows
  - TCP Slow Start gets slower!
- Limited Throughput
  - 32K per RTT protocol limitation
- Slow reaction to congestion levels

Delay mitigation strategies

- Avoid it in the first place
- Mitigate the delay source

Avoid delay in the first place

- Bring the data source closer to the consumer
- Local caches
  - FTP “mirror sites”
  - Web caches
- Local services
  - Local computation

Obvious sources of delay

- Router Queuing delay
- Transmission Propagation delay
- “Window depleted” period
  - where sender is blocked by receiver

Mitigate queuing delays

- Queuing algorithms
  - FIFO Queuing
  - Class-based Queuing
  - Weighted Fair Queuing
- Line disciplines
  - PPP fragmentation
  - Multiplexed ATM VCs

Mitigate propagation delays

- These result from physics
  - Which mere mortals can't readily change
- Can we parallelize the system?
  - Long windows + Selective Acknowledge
  - Parallel file transfers
- Can we make good use of recent history?
  - HTTP 1.1 persistent TCP connections

Mitigate “window depleted” intervals

- TCP behavior
- Traffic rate controls
- Overlapping windows with rate-based controls

TCP behavior

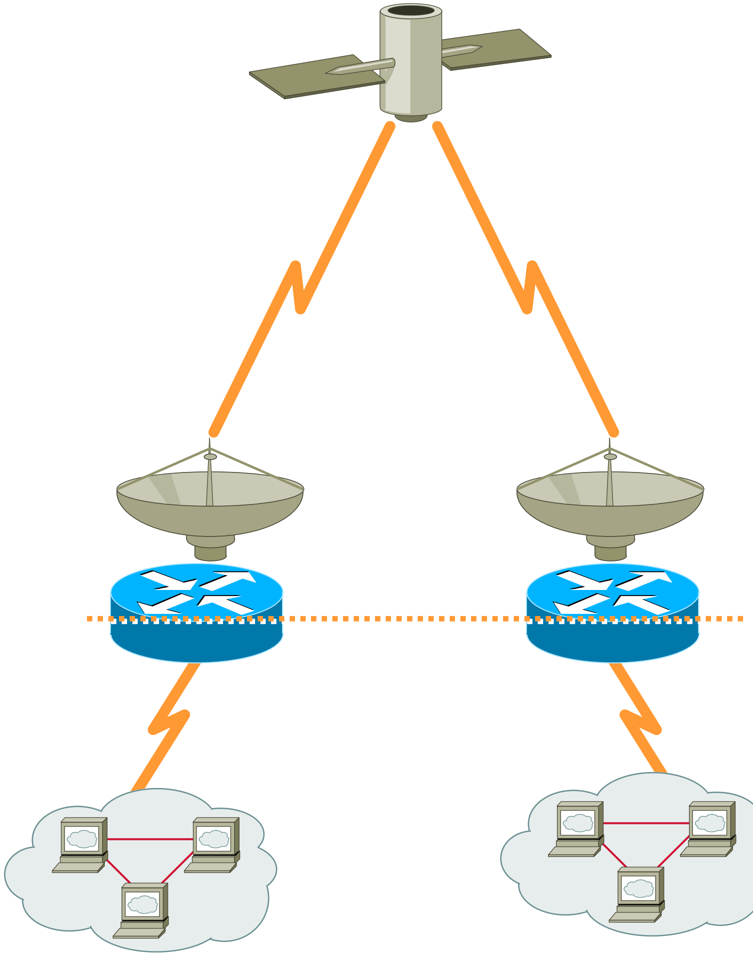
- Slow start
- Fast retransmission
- Traffic drops seen as indications of congestion

Traffic rate controls

- Sliding window controls
  - Constant data outstanding
- Rate based controls
  - Constant transmission rate

Subdivide the TCP connection

- TCP at each end
- Reliable link between points
  - Could be TCP, not required
  - Limit each connection to rate supported by next connection
  - Large effective window



Net effect:

- Throughput governed by slowest connection
  - High delay connection has pseudo-rate based control governed by slow start at endpoints
- Duration of data transfer:
  - Duration of transfer disregarding propagation delay
  - Plus 1-2 round trip delays

Quality of Service

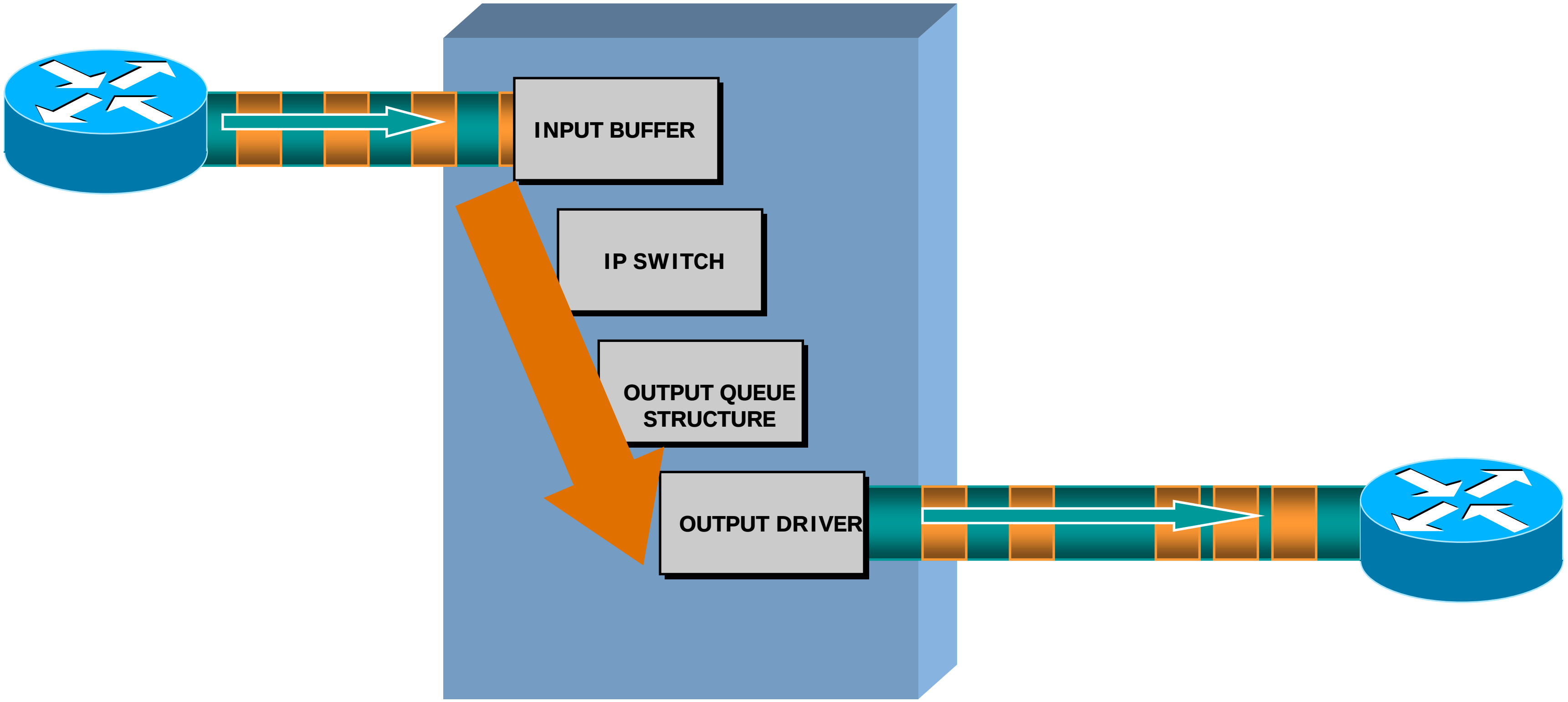
Approaches to Quality Management

What are the Q variables?

- A network is composed of a set of
  - routers
  - switches
  - other network attached devices

  Connected by
  - transmission links

Router Service Components

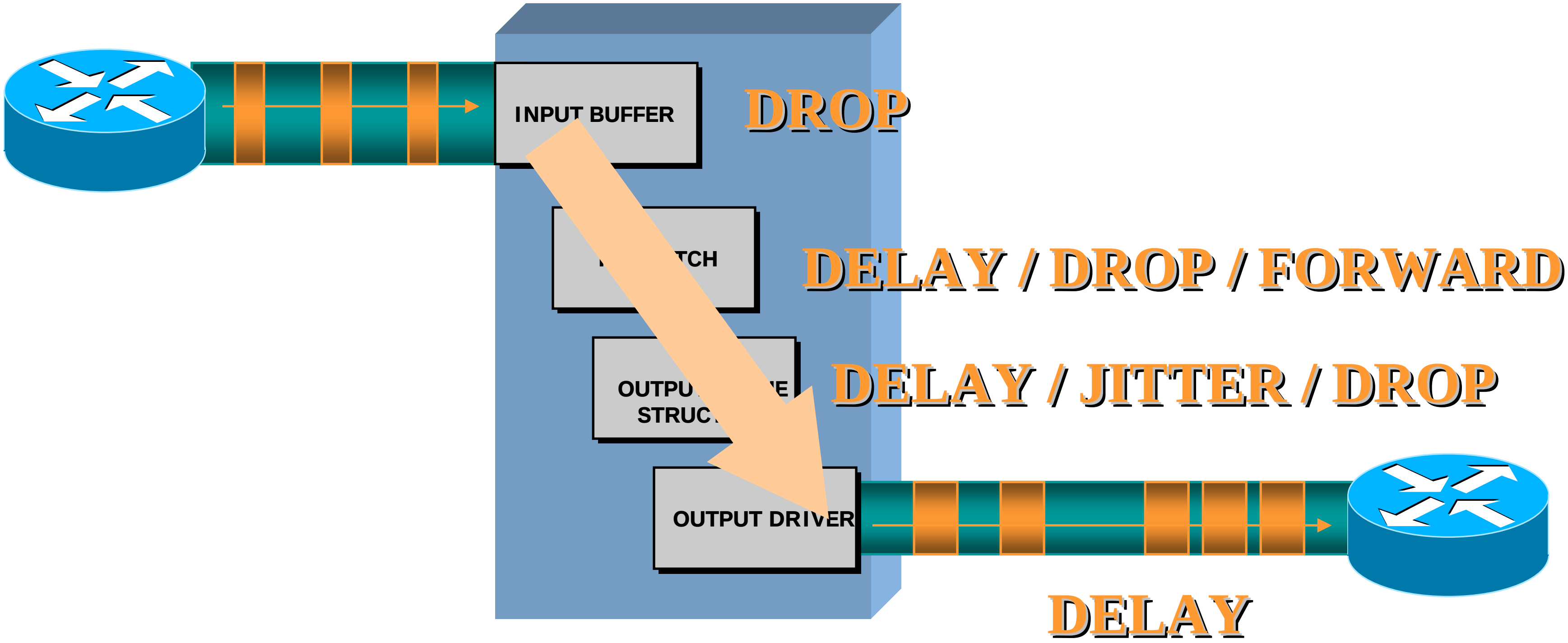


Router Q variables?

- Routers can:
  - fragment
  - delay
  - discard
  - forward

packets through manipulation of ingress & queue management and forwarding mechanisms

Router Q variables



Transmission Q variables

- Constant flow point-to-point bit pipes
  - constant delay
  - packet loss probability related to transmission error rate and link MTU
- No intrinsic differentiation on loss and delay is possible

Transmission Q variables

- Switched L2 services:
  - ATM, Frame Relay, SMDS
  - create virtual end to end circuits with specific carriage characteristics
- Variable delay and loss probability is possible

Transmission Q variables

- Multiple access LANs
  - variable delay and loss probability based on access algorithm, which is effected by imposed load
- No predetermined differentiation on loss and delay is possible
  - although some efforts are underway to change this for LAN technologies

How to differentiate flows

- Use **state-based** mechanisms to identify flows of traffic which require per-flow differentiation
- Use **stateless** mechanisms that react to marked packets with differentiated servicing

Integrated Services

- Per flow traffic management to undertake one of more of the following service commitments:
  - Place a preset bound on jitter.
  - Limits delay to a maximal queuing threshold.
  - Limit packet loss to a preset threshold.
  - Delivers a service guarantee to a preset bandwidth rate.
  - Deliver a service commitment to a controlled load profile.
- Challenging to implement in a large network.
- Relatively easy to measure success in meeting the objective.

IntServ and the Internet

- RSVP approach
- Integrated Services requires the imposition of flow-based dynamic state onto network routers in order to meet the stringent requirements of a service guarantee for a flow.
- Such mechanisms do not readily scale to the size of the Internet.

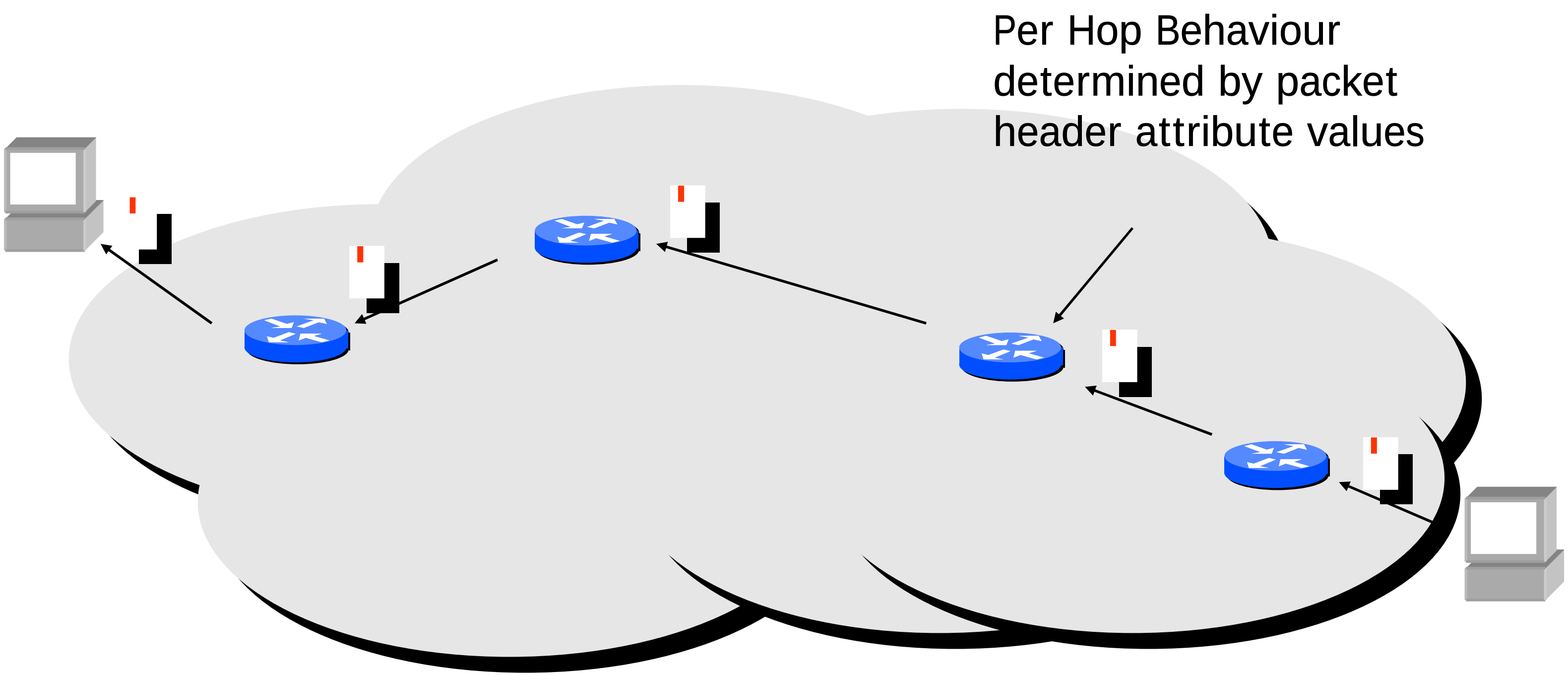
Differentiated Services

- Active differentiation of packet-based network traffic to provide a *better than best effort* performance for a defined traffic flow, as measured by one of more of:
  - Packet jitter
  - Packet loss
  - Packet delay
  - Available peak flow rate
- Implementable within a large network.
- Relatively difficult to measure success is providing service differentiation.

DiffServ and the Internet

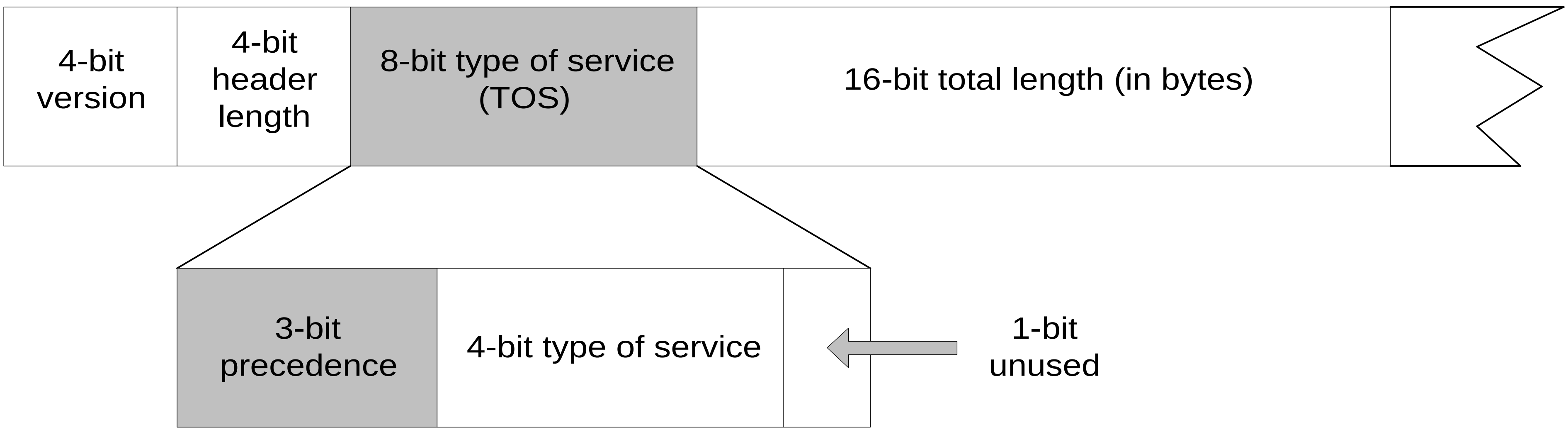
- Approach being considered within IETF
- Differentiated Services can be implemented through the deployment of differentiation router mechanisms triggered by per-packet flags, preserving a stateless network architecture within the network core.
- Such mechanisms offer some confidence to scale to hundreds of millions of flows per second within the core of a large Internet

Stateless QoS



Diffserv currently discussing use of TOS byte

IP Header (first 32 bits)



8-bit type of service (TOS)

Quality of Service

Considerations

“Managed Expectations Internet”

- Solve congestion problems with
  - TCP implementation improvements
  - Weighted Random Early Detection
- Manage users to a contracted rate
- Permit use of excess when available

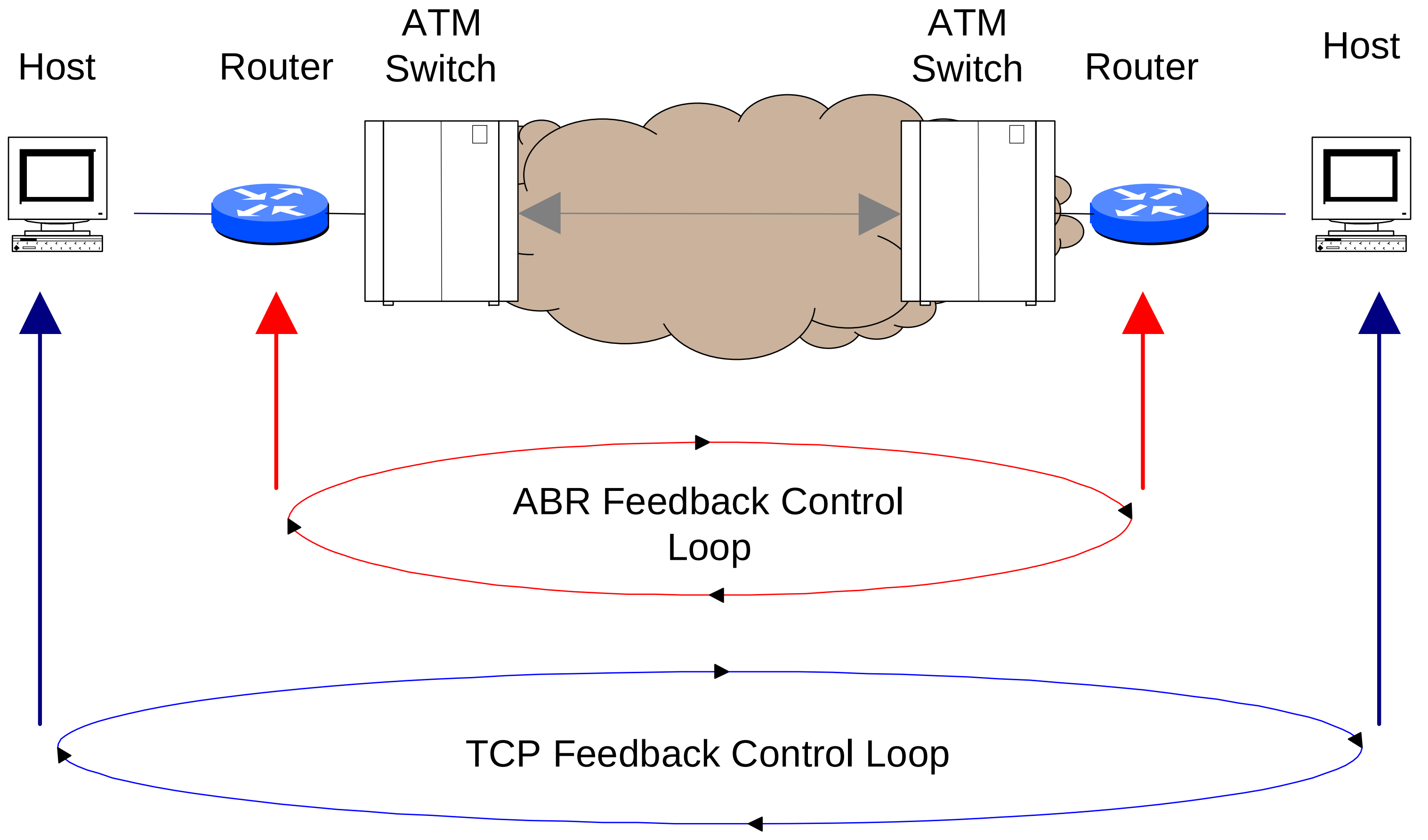
Service contracts in the "new" Internet ?

- Tiered cost structure
  - Low cost for contracted service
  - Additional cost for excess service
  - Additional cost for specialized calls
- QoS Routing could support "specialized calls"

End-to-End QoS

- Reliance on a particular link-layer technology to deliver QoS is fundamentally flawed.
- TCP/IP is the “common bearer service,” the most common denominator in today’s Internet.
- Partial-path QoS mechanisms introduce distortion of the data flow and are ineffectual.
- Must scale to hundreds of thousands of active flows, perhaps millions.

Again: What is End-to-End?



Pervasive homogeneity - Not!

- Reliance on link-layer mechanisms to provide QoS assumes pervasive end-to-end, desktop-to-desktop, homogenous link-layer connectivity.
  - This is simply not realistic.
- QoS as a differentiation mechanism will be operated in an variable load environment
  - differentiation will be non-repeatable

State and Scale

- To undertake firm commitments in the form of per-flow carriage guarantees requires network-level state to be maintained in the routers.
- State becomes a scaling issue.
- Wide-scale RSVP deployment will not scale in the Internet (See: RFC2208, RSVP Applicability Statement).

Network Layer Tools

- Traffic shaping and admission control.
- IP packet marking for both delay indication & discard preference.
- Weighted Preferential Scheduling algorithms.
- Preferential packet discard algorithms (e.g. Weighted RED, RIO).

- End result: Varying levels of best effort under load.
- No congestion, no problem. (Well, almost.)

Quality of Service

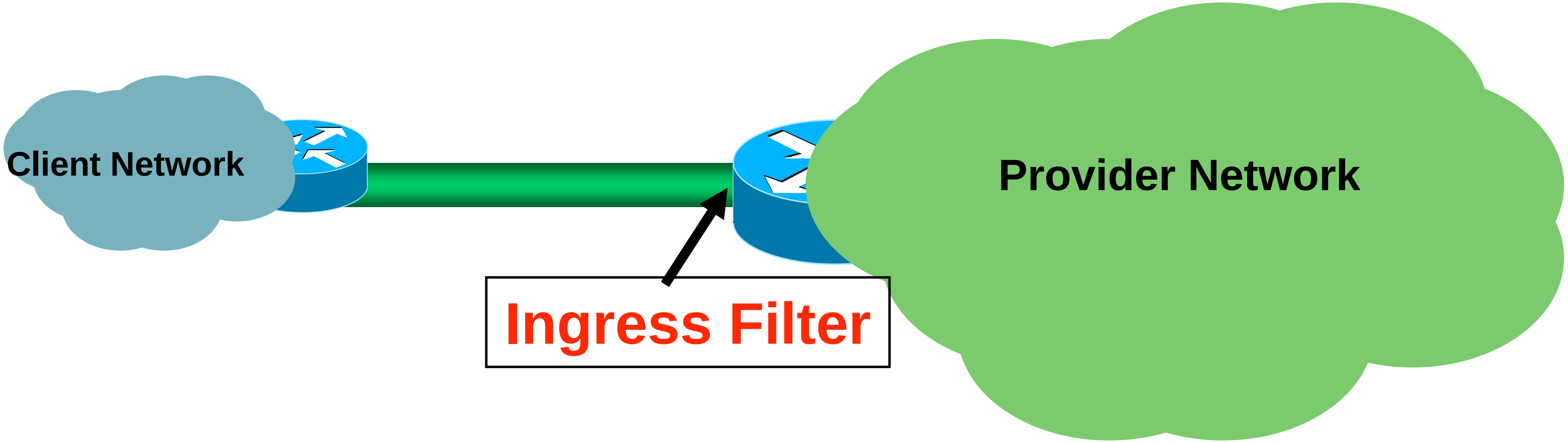
Mechanisms

Router Mechanisms

- A router has a limited set of QoS responses:
  - Fragmentation
    - fragment the packet (if permitted)
  - Forwarding
    - route the packet to a particular interface
  - Scheduling
    - schedule the packet at a certain queuing priority, or
    - discard the packet

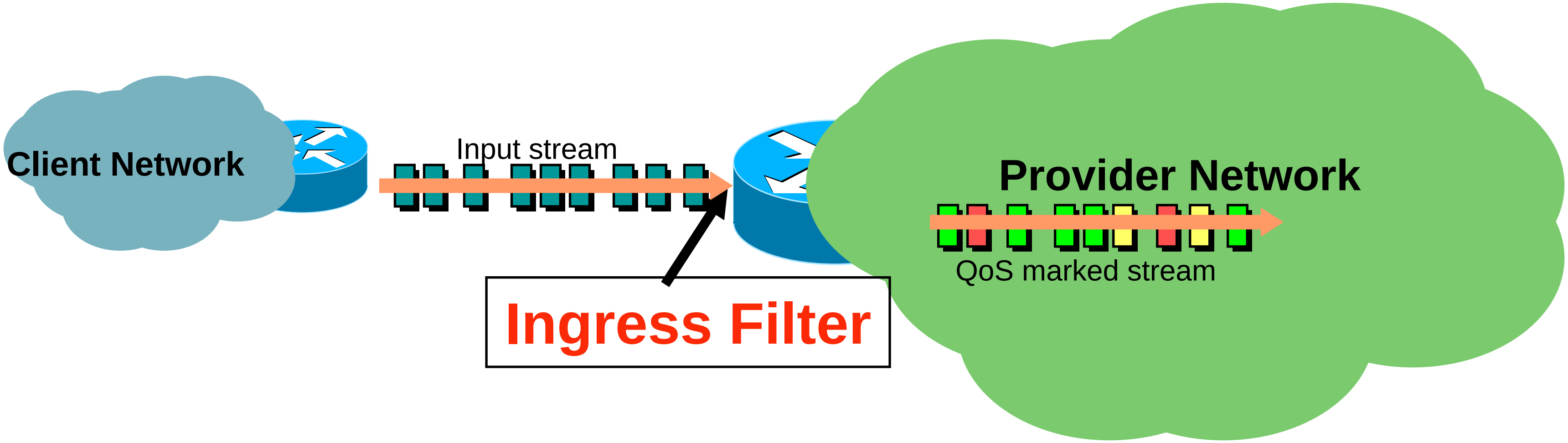
QoS Service Element

- What is the service element for QoS services?
  - Network ingress element



QoS Service mechanism

- Admission traffic profile filter
  - In-Profile traffic has elevated QoS, out-of-profile uses non-QoS



QoS Service by Application

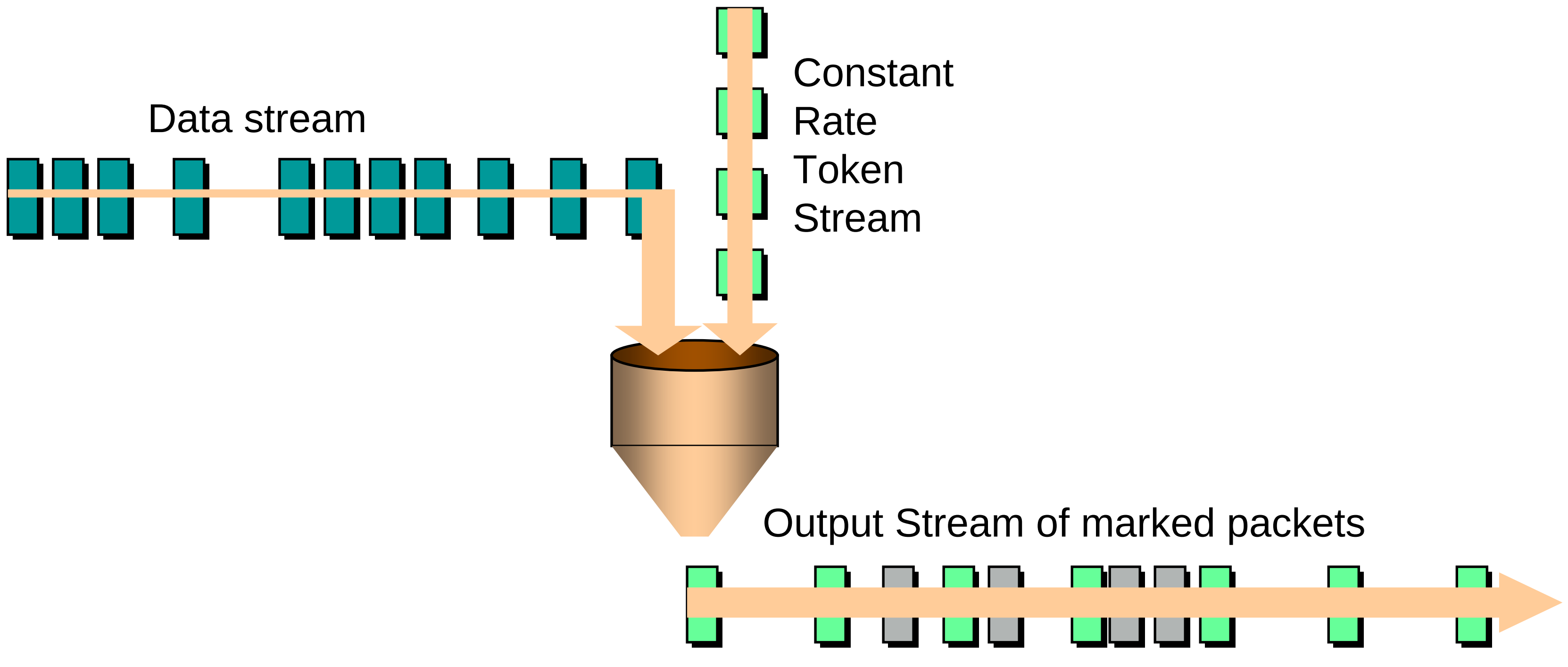
- Application-based differentiation at ingress:
  - TCP or UDP port number
  - For example:
    - set elevated priority for interactive services
      - ports 80, 23, 523
    - set background priority for bulk batch services
      - port 119

QoS Service by Host

- Selected sender’s traffic has elevated QoS:
  - Source IP address filter
    - If source address matches a.b.c.d set precedence to *p*
- Traffic to selected receiver has elevated QoS
  - Destination IP address filter
    - if destination address matches a.b.c.d set precedence to *p*
- Traffic between selected sender and receiver has elevated QoS
  - Flow based QoS, using dynamic selection of an individual flow
  - flow-class based QoS, triggered by source, receiver and port mask

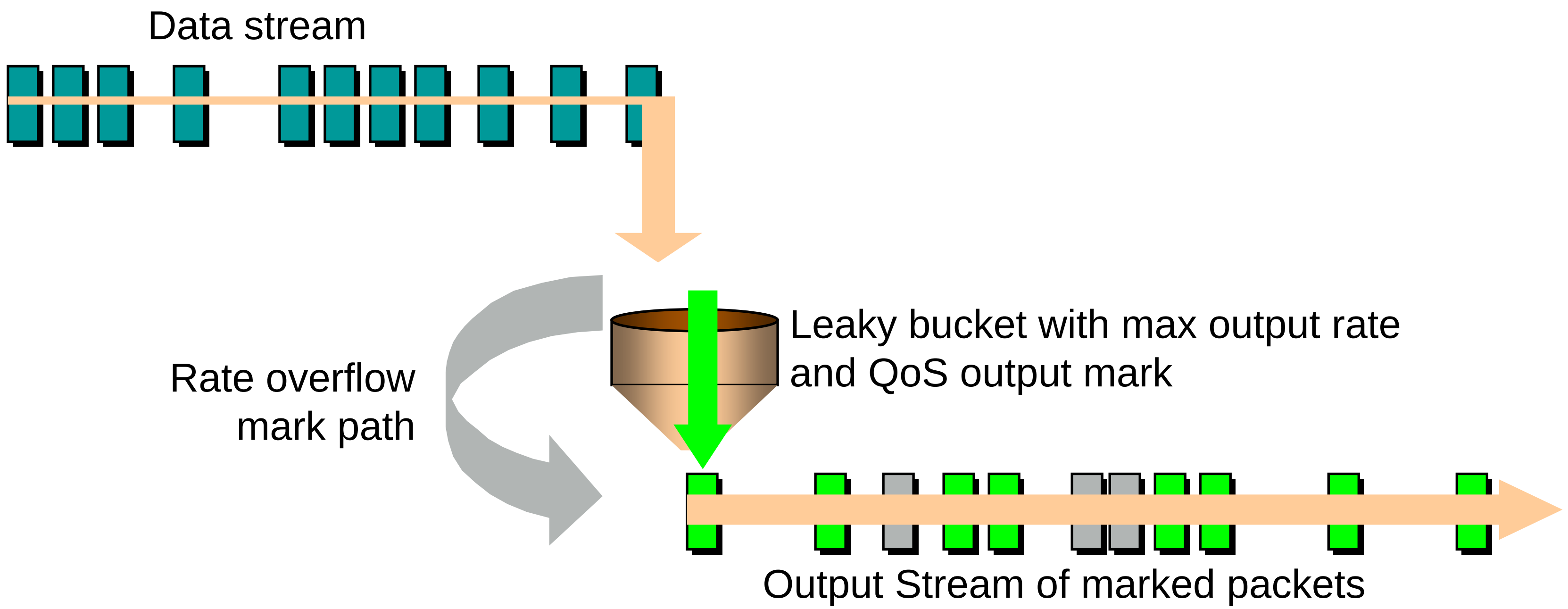
QoS Service by profile

- profile based on *token bucket* for constant rate profile



QoS Service by profile

- profile based on *leaky bucket* for controlled burst profile



QoS Admission model

- Network defined and determined
  - user QoS indicators are cleared at ingress, replaced by network defined QoS indicators
- User selected, network filters
  - User QoS tags are compared against contracted profile of admitted traffic
    - within contract packets are admitted unchanged
    - other packets have cleared QoS indication

Precedence selection based on contract

- Network enforces precedence value
  - Source or Destination Address
  - ingress interface
- Might have two or three precedence values
  - such as

    1 - RSVP traffic

    2 - Best effort traffic within some token bucket

    3 - All other traffic

QoS per packet indicators

- Explicit per packet signaling of:
  - Precedence indication (delay)
  - Discard indication (reliability)

As an indication of preference for varying levels of best effort.

- This is deployable - today.

QoS Indicators

- How to ingress mark a QoS packet?
  - Precedence marking
    - use a precedence value field in the packet header
    - field value triggers QoS mechanisms within the interior of the network
  - Drop Preference marking
    - use a drop preference value filed in the packet header
    - interior switches discard packets in order of drop preference

QoS Indicators

- both precedence and drop preference require uniform responses from the interior of the network, which in turn requires:
  - uniform deployment of QoS-sensitive mechanisms within routers
  - uniform inter-provider mechanisms (where agreed)

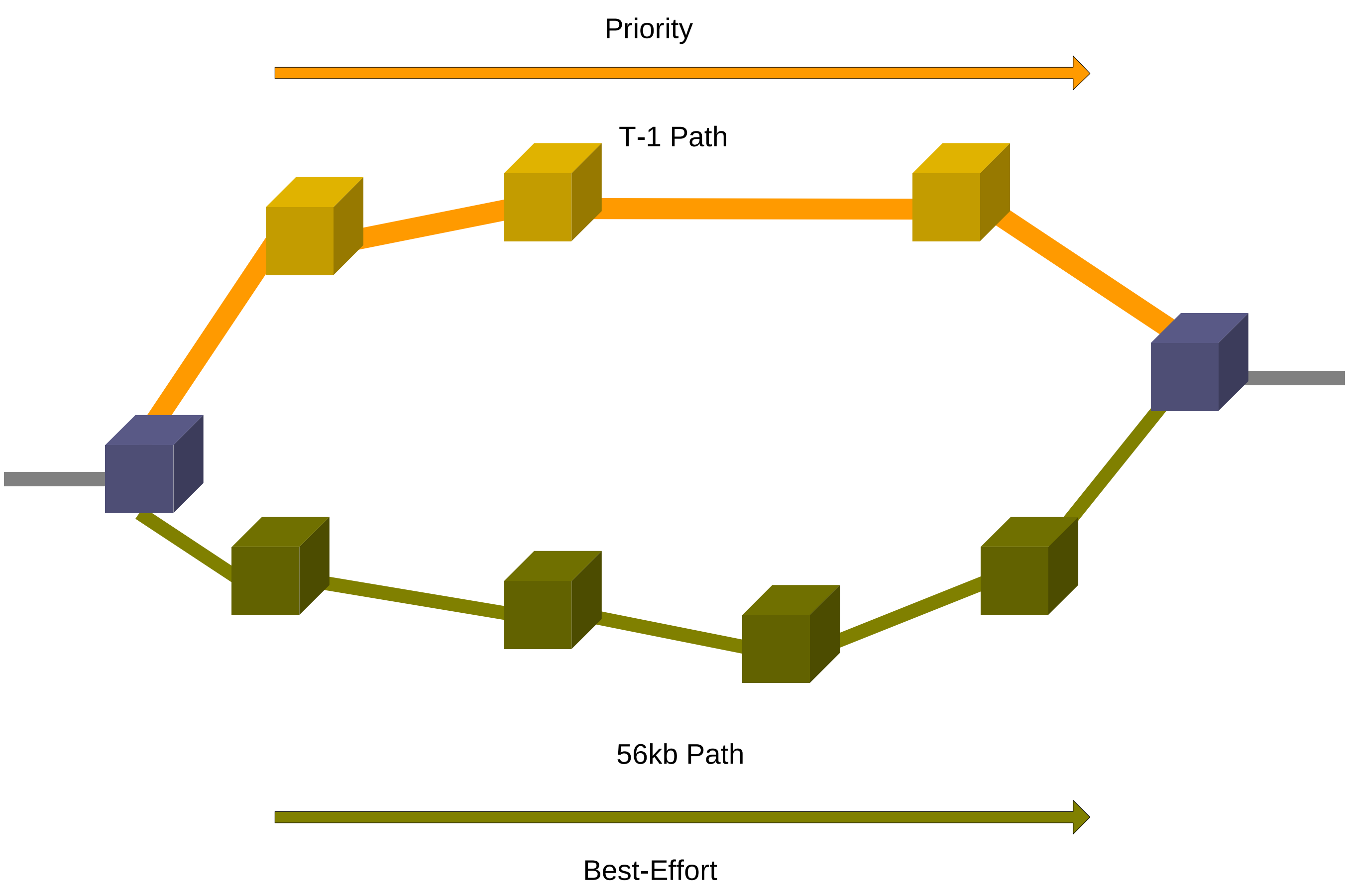
Virtual Circuits

- Segmented bandwidth resource for QoS states:
  - Ingress traffic shaping (token or leaky bucket)
  - Virtual circuits & statistical muxing (e.g. ATM, Frame Relay)
  - RSVP admission control & reservation state
- Circuit segmentation mechanisms by themselves are unrealistic in a large scale heterogeneous Internet which uses end-to-end flow control.

QoS Paths

- Alternate path selection
  - Alternative physical paths
    - E.g., cable and satellite paths
  - QoS Routing v. administrative path selection
- Must be managed with care.
- Can lead to performance instability.
- Prone to inefficient use of transmission.
- May not support end-to-end path selection

Alternate paths



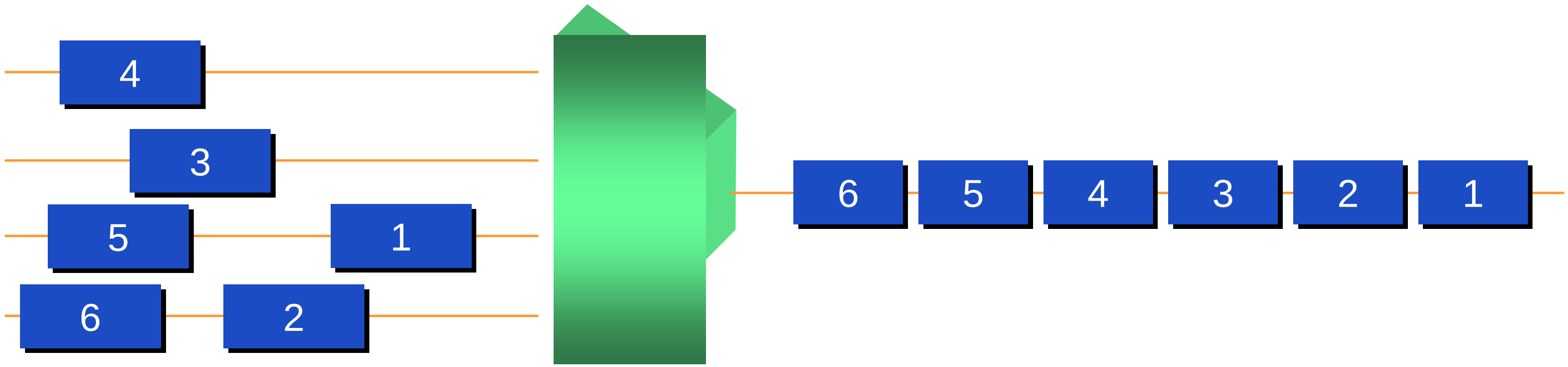
Quality of Service

Queuing Disciplines

FIFO queuing

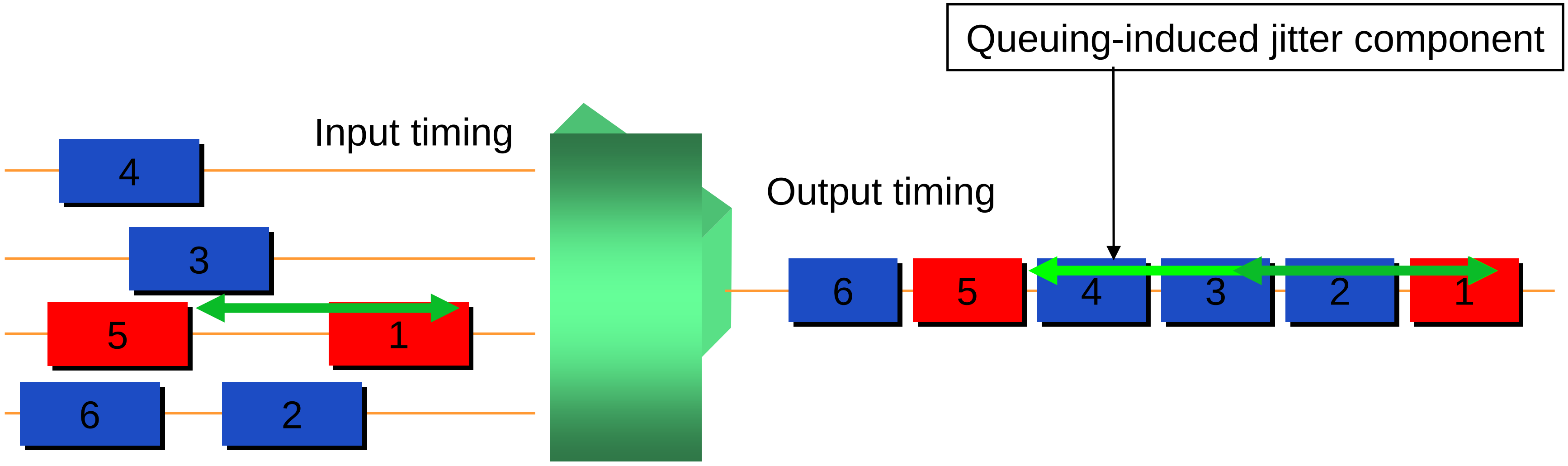
FIFO queuing

- Strict Round-Robin queuing discipline



Effects of FIFO queuing

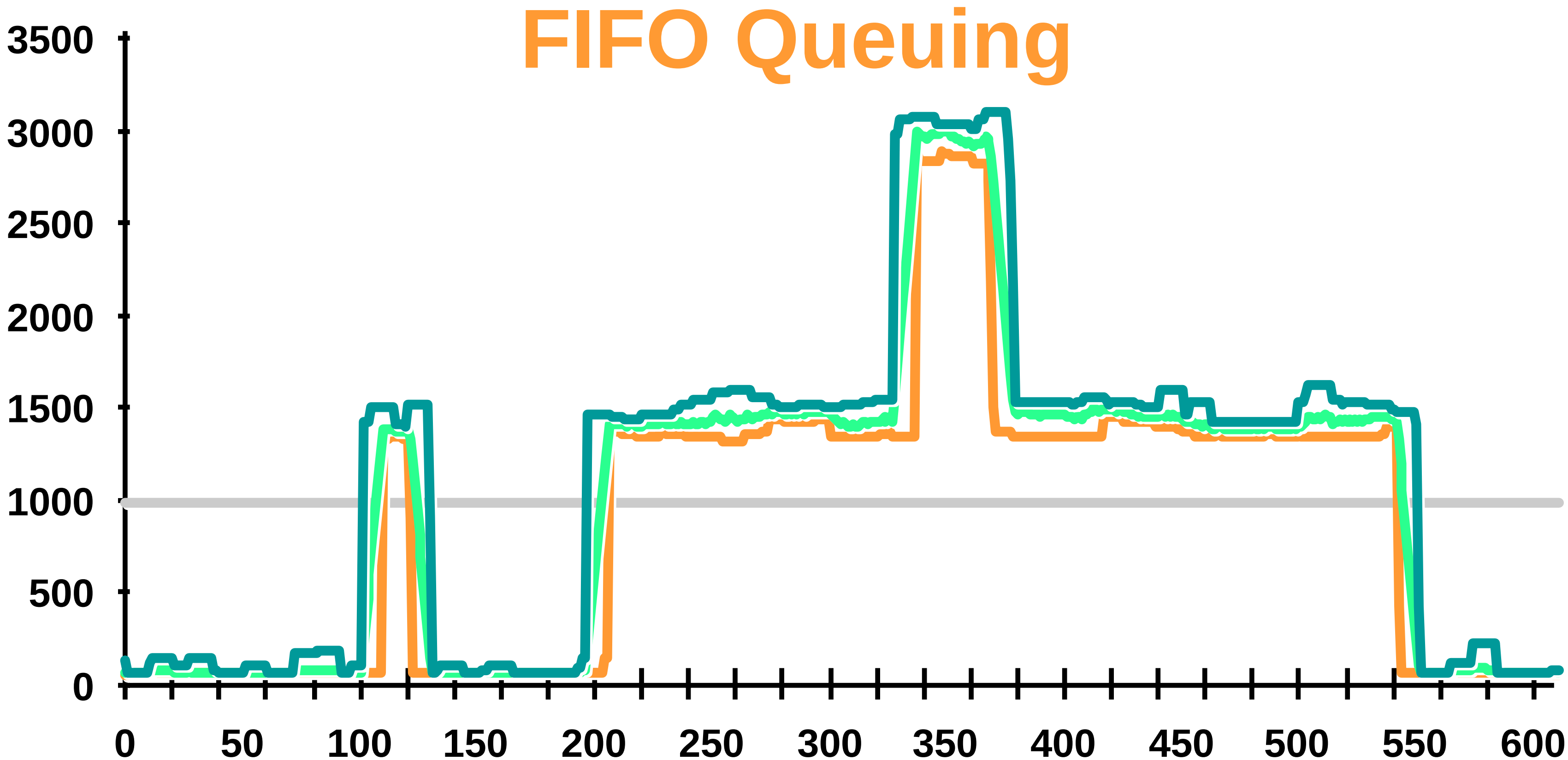
- Packet trains:
  - Delay
  - Jitter



Effects of FIFO queuing

- Tail drop yields performance collapse
  - FIFO queuing causes queue pressure, resulting in queue exhaustion and tail drop
  - tail drop causes packet trains to have trailing packets discarded
  - Without following packets the receiver will not send duplicate ACKs for the missing packets
  - The sender may then have to timeout to re-transmit
  - The timeout causes the congestion window to close back to a value of 1, and restart with Slow Start rate control

Interactive Traffic Timing



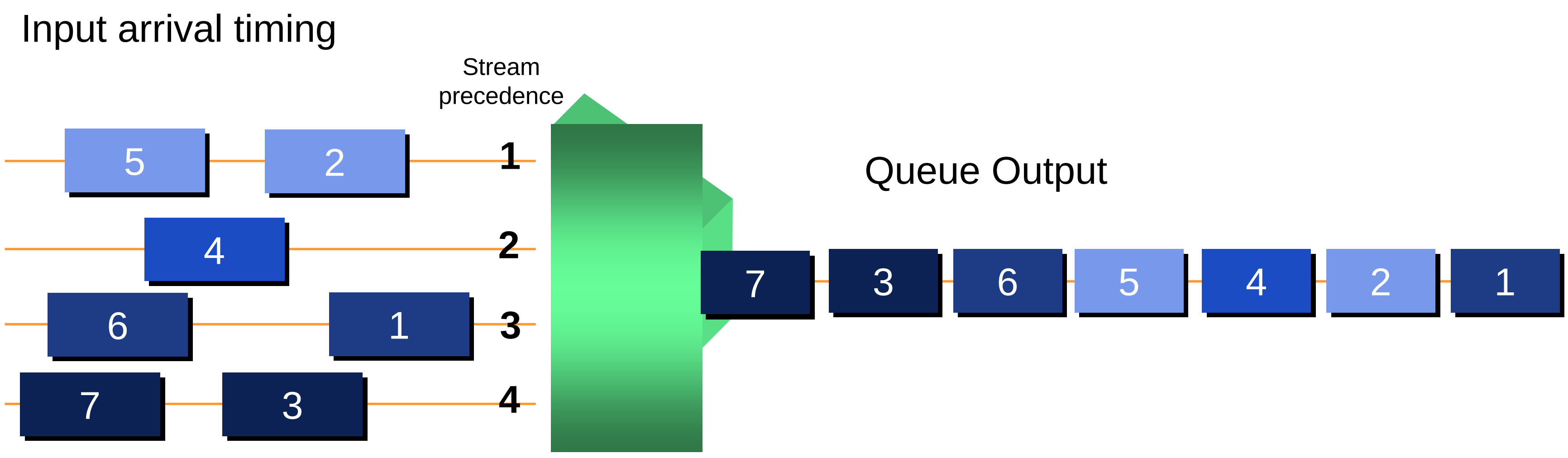
Quality of Service

Queuing Disciplines

Precedence queuing

Precedence Queuing

- multiple queues, each served in FIFO order



Precedence Queuing

- Jitter and delay for high precedence queues still present at short time intervals
- Precedence algorithm denies any service to lower level queues until all higher level queues are exhausted
  - This allows high precedence TCP sessions to open up sending window to full transmission capacity
  - this causes protocol collapse for lower layer queues

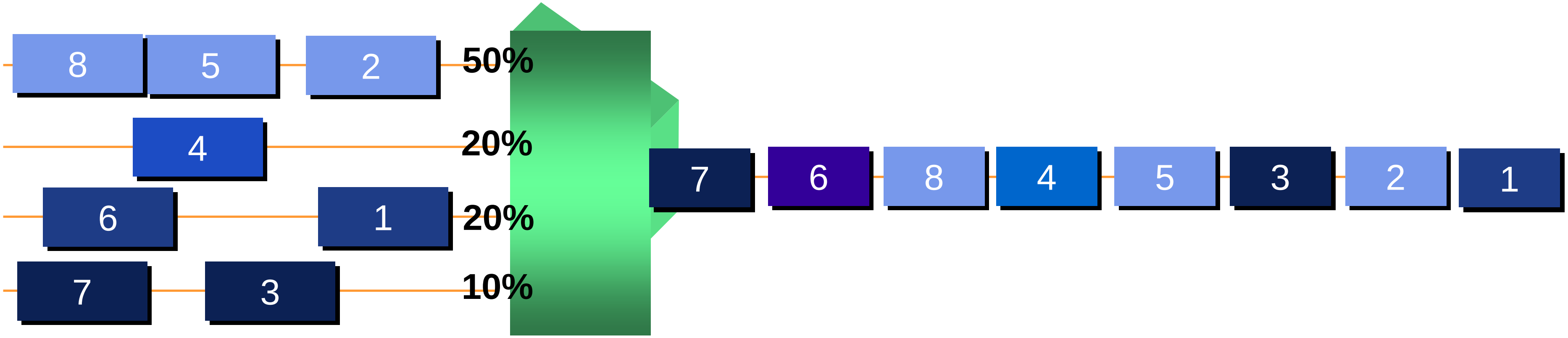
Quality of Service

Queuing Disciplines

Class-based queuing

Class-based Queuing

- multiple queues, serviced in proportionate levels



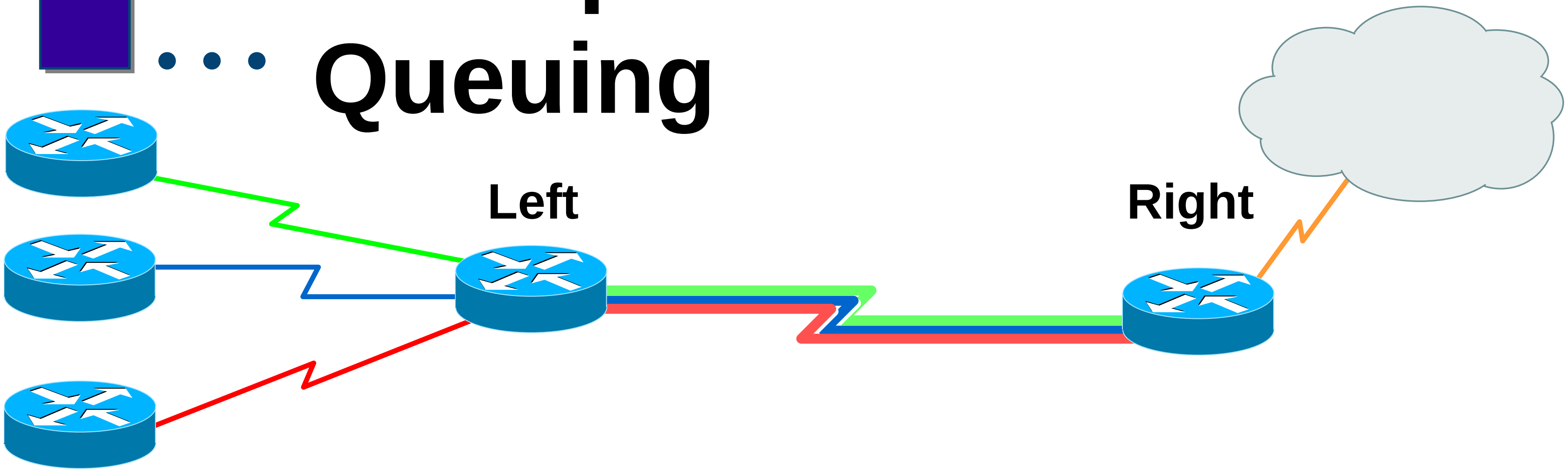
Class-based queuing

- Divide service among traffic classes
- Divide service among delay classes

Class-based Queuing

- Class-based queues attempt to allocate fixed proportion of resource to each service queue
- Address denial of service by attempt to guarantee some level of service is provided to each queue
- Class-based queues are an instance of a more general proportionate sharing model
- Class-based queues are fair only for time intervals greater than number of service classes multiplied by link MTU transmission time

Example of Class-based Queuing



Left router

- Queue-list by incoming interface
- Bytes per MTU rotation proportional to fiscal input

Right router

- Queue-list by destination CIDR prefix
- Bytes per MTU rotation proportional to fiscal input

Quality of Service

Queuing Disciplines

Weighted Fair Queuing

Weighted Fair Queuing

- Attempts to schedule packets to closely match a theoretical bit-wise weighted min-max allocation mechanism
- The mechanism attempts to adhere to the resource allocation policies at time scales which are finer than class-based queuing

Min-Max weighted fairness

- Allocate resources to each stream in accordance with its relative weight, and interatively redistribute excess allocation in accordance with relative weight

Initial Weighting

Stream	Weight	Allocation
A	5	42%
B	3	25%
C	3	25%
D	1	8%

Re-allocation following stream termination, where 25% is redistributed to the remaining streams in the ration 5:3:1

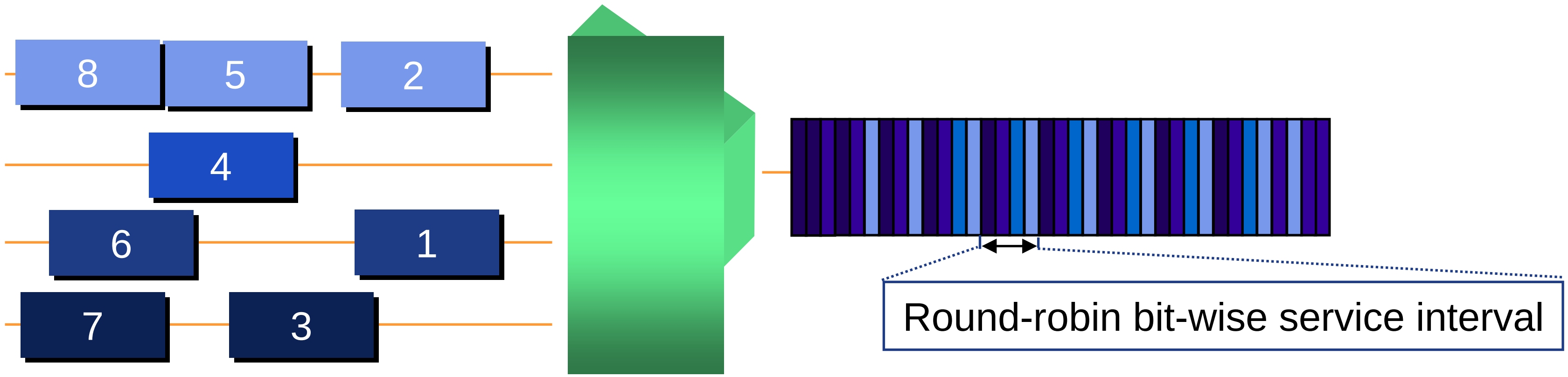
Stream	Weight	Allocation
A	5	56%
B	3	33%
D	1	11%
C	inactive	

Bit-wise Round Robin Fair Queuing

- Fair Queuing Objectives:
  - Simulates a TDM
  - One flow per TDM virtual channel

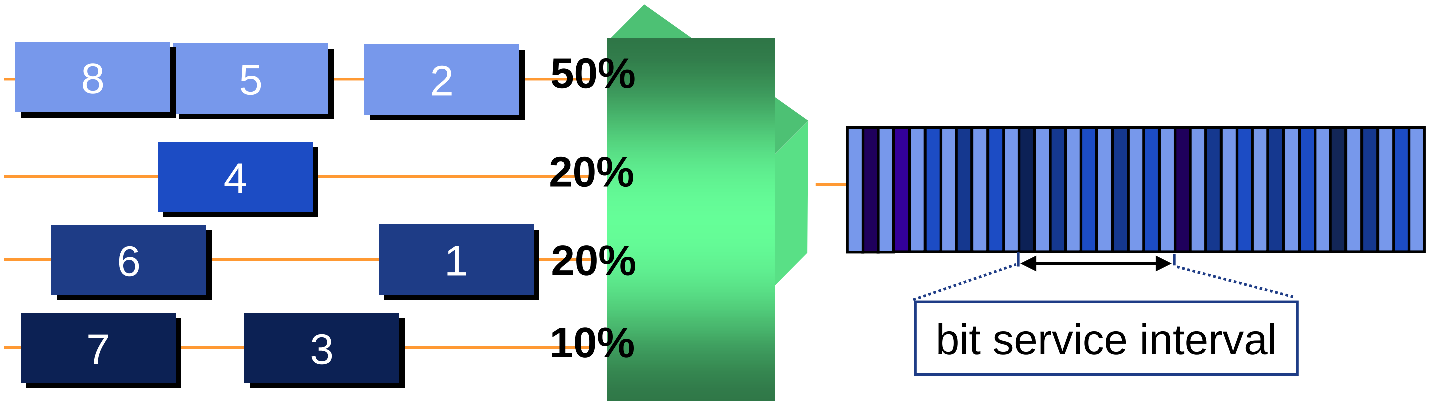
Bit-wise Scheduling

- Bits from each class are serviced in strict rotation
- This is equivalent to Time Division Multiplexing



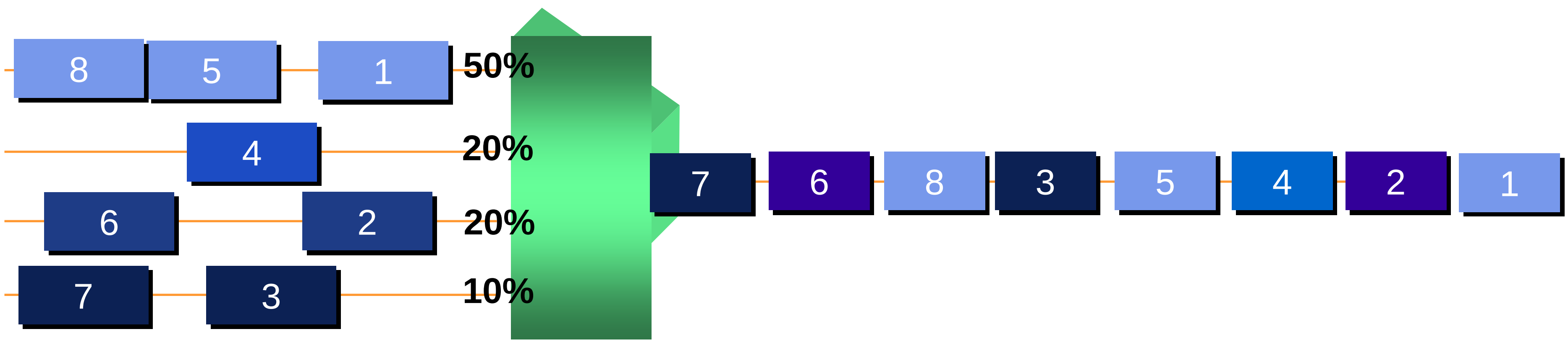
Weighted Bit-wise Scheduling

- bits from each class are serviced in rotation, weighted by relative service weight



Weighted Fair Queuing

- Schedule traffic in the sequence such that a equivalent weighted bit-wise scheduling would deliver the same order of trailing bits of each packet



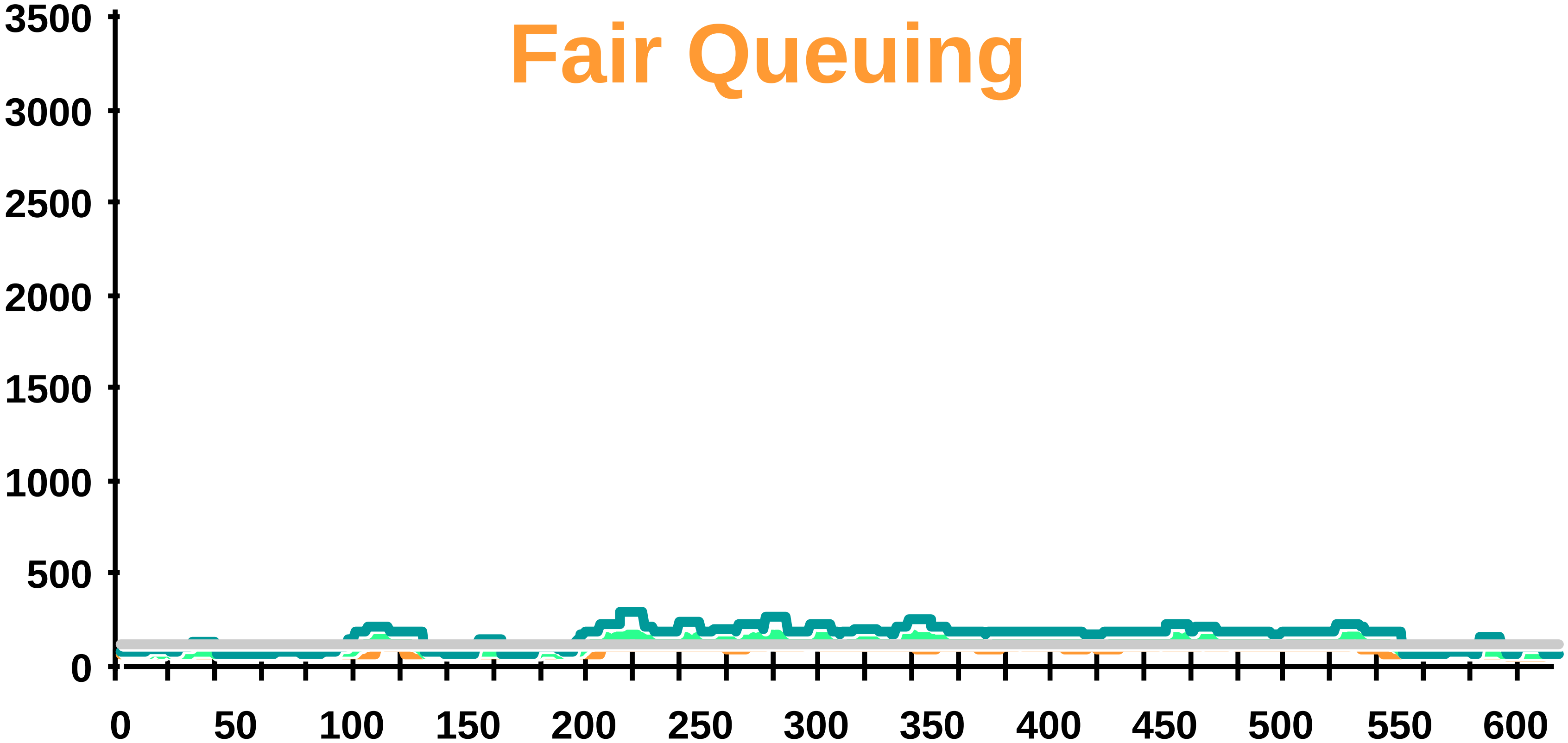
Weighted Fair Queuing

- As a result, be as fair as weighted bit-wise scheduling, modulo packet quantization
- Weighted fair queuing is min-max fair
- Weighted fair queuing does require extensive processing to determine the weighted TDM finish time of each packet
- Weighted fair queuing scales per precedence level, and not necessarily on a per flow basis.

Weighted Fair Queuing

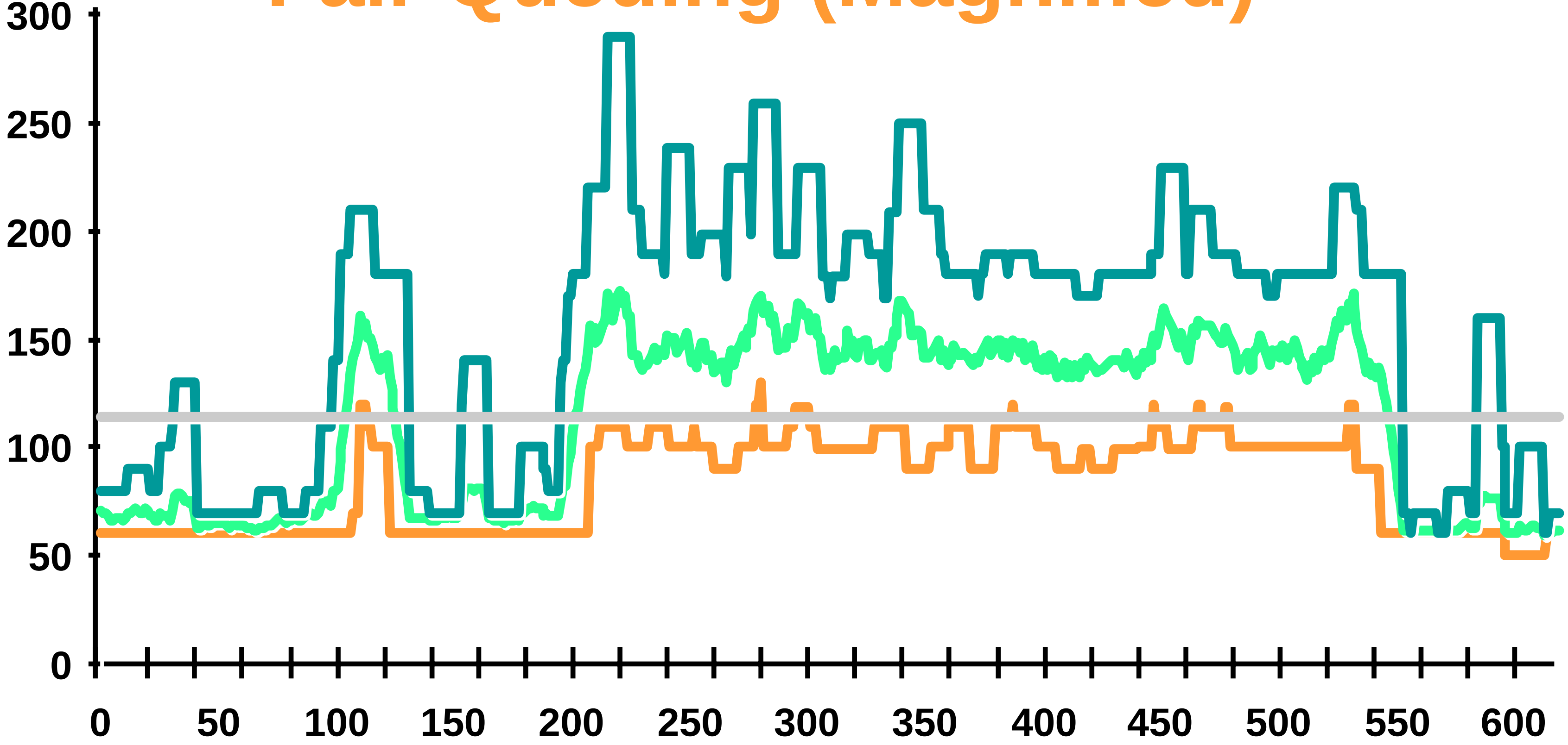
- Low queue occupancy flows
  - All the bandwidth they can use
  - Minimal delay
- High queue occupancy flows
  - Enforce traffic interleaving
  - Fair throughput rates

Interactive Traffic Timing



Interactive Traffic Timing

Fair Queuing (Magnified)



Weighted Fair Queuing

- Appropriate when
  - Important flows have significant amounts of data in queue
- Can be used for various administrative models
- Traffic classification based on
  - Source/destination information
  - per data flow
  - Artifacts of network engineering
  - packet indication (precedence value)

Quality of Service

Queue Management

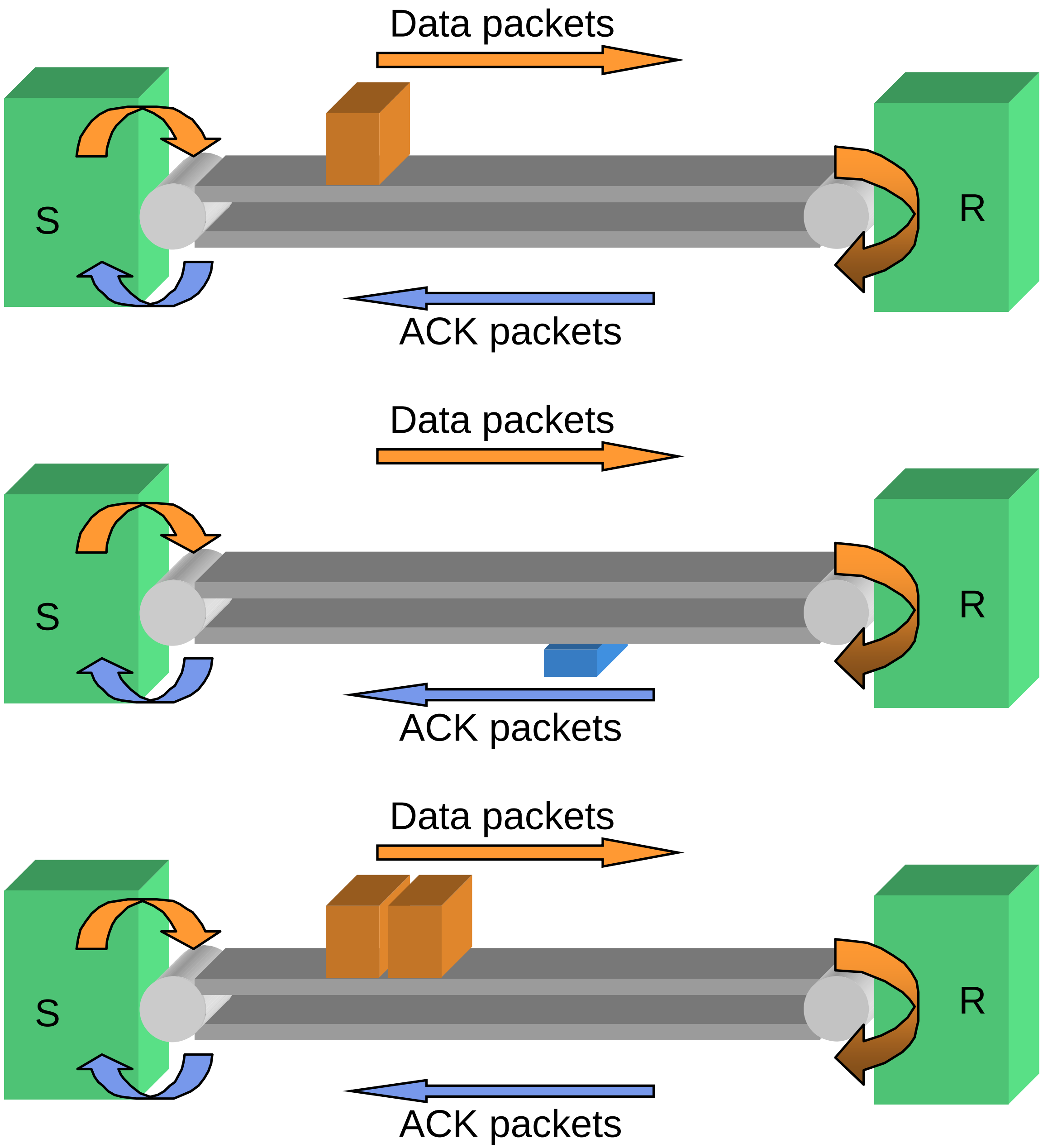
Weighted Random Early Deletion

Weighted Random Early Deletion - W-RED

- Stated requirement
  - “Avoid congestion in the first place”
  - “Statistically give some traffic better service than others”
- Congestion avoidance, rather than congestion management

Behavior of a TCP Sender

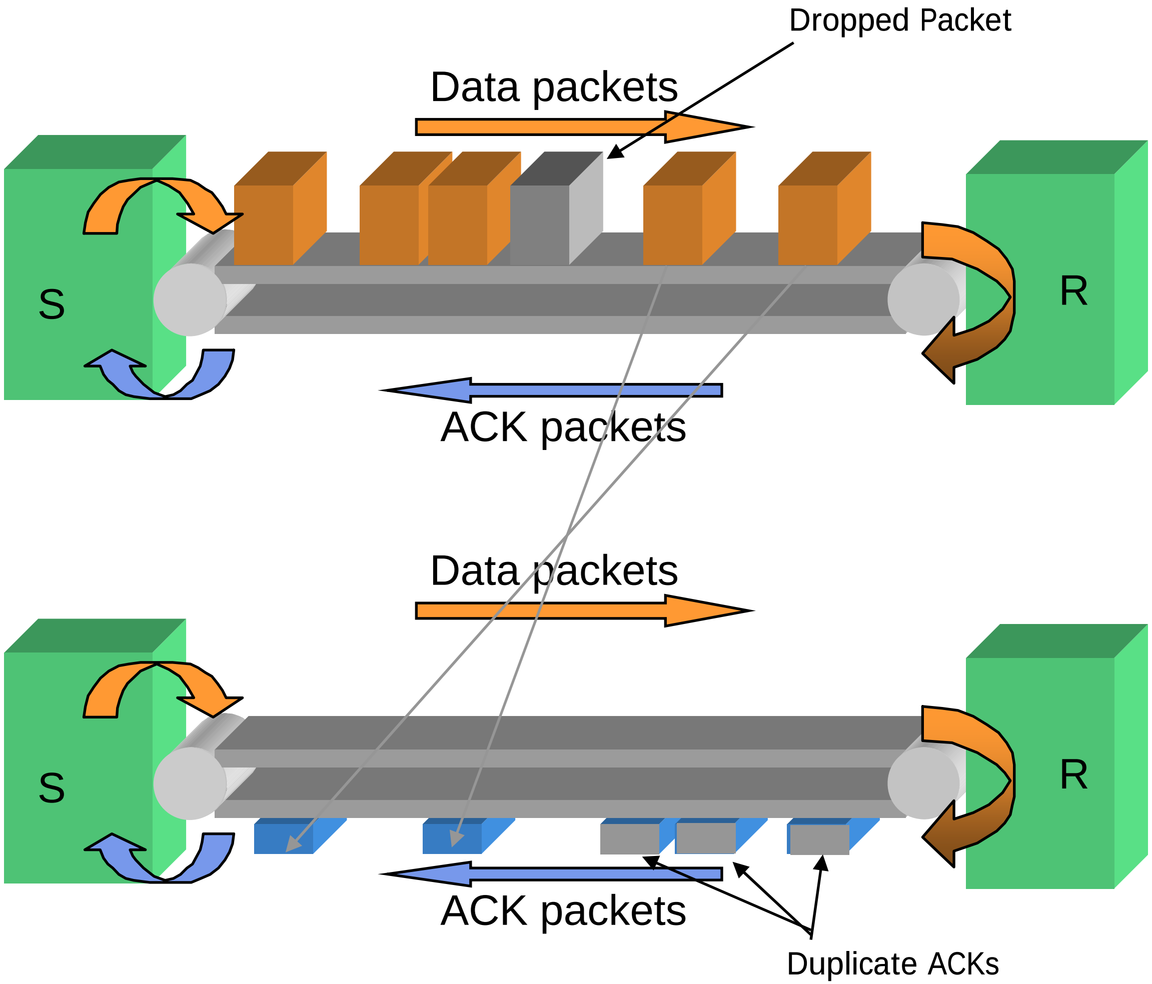
- Sends as much as credit (TCP window) allows
- Starts credit small (initial *cwnd* = 1)
  - Avoid overloading network queues
- Increases credit exponentially (slow start) per RTT
  - To gauge network capability via packet loss signal



Data packets
S
R
ACK packets
Data packets
S
R
ACK packets
Data packets
S
R
ACK packets

Behavior of a TCP Receiver

- When in receipt of “next message,” schedules an ACK for this data
- When in receipt of something else, acknowledges all received in-sequence data immediately
  - i.e. send duplicate ACK in response to out of sequence data received


Dropped Packet
Data packets
S
R
ACK packets
Data packets
S
R
ACK packets
Duplicate ACKs

Sender Response to ACK

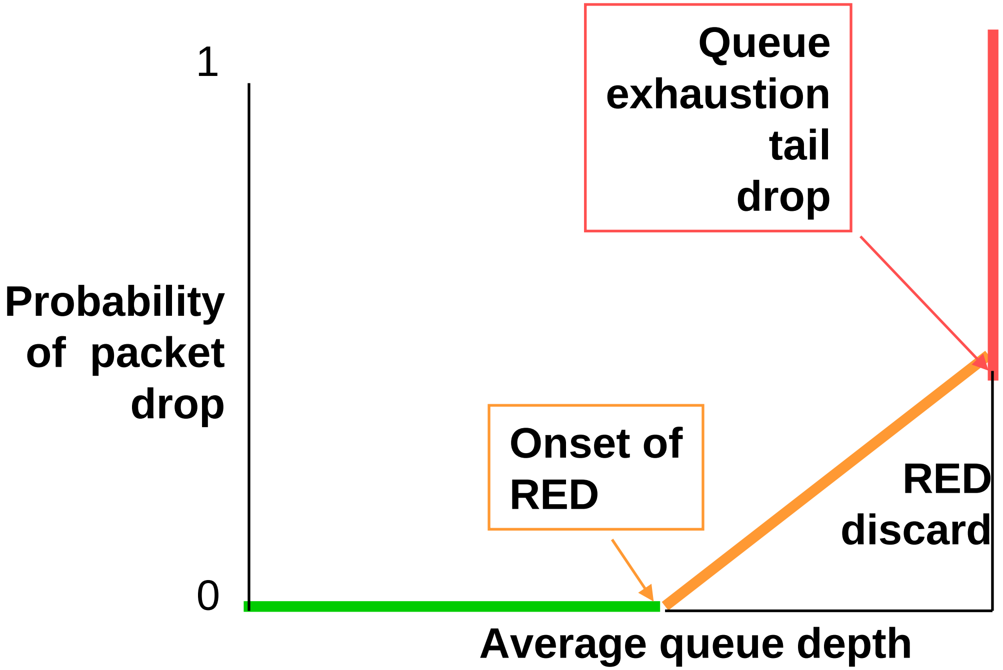
- If ACK advances sender's window
  - Update window and send new data
- If not then it's a duplicate ACK
  - Presume it indicates a lost packet
  - Send first unacknowledged data immediately
  - Halve current sending window
  - shift to congestion avoidance mode
  - Increase linearly to gauge network throughput

Implications for Routers

- Dropping a data packet within a data sequence is an efficient way of indicating to the sender to slow down
  - Dropping a data packet prior to queue exhaustion increases the probability of successive packets in the same flow sequence being delivered, allowing the receiver to generate duplicate ACKs, in turn allowing the sender to adjust *cwnd* and reducing sending rate using fast retransmit response
  - Allowing the queue to fill causes the queue to tail drop, which in turn causes sender timeout, which in turn causes window collapse, followed by a flow restart with a single transmitted segment

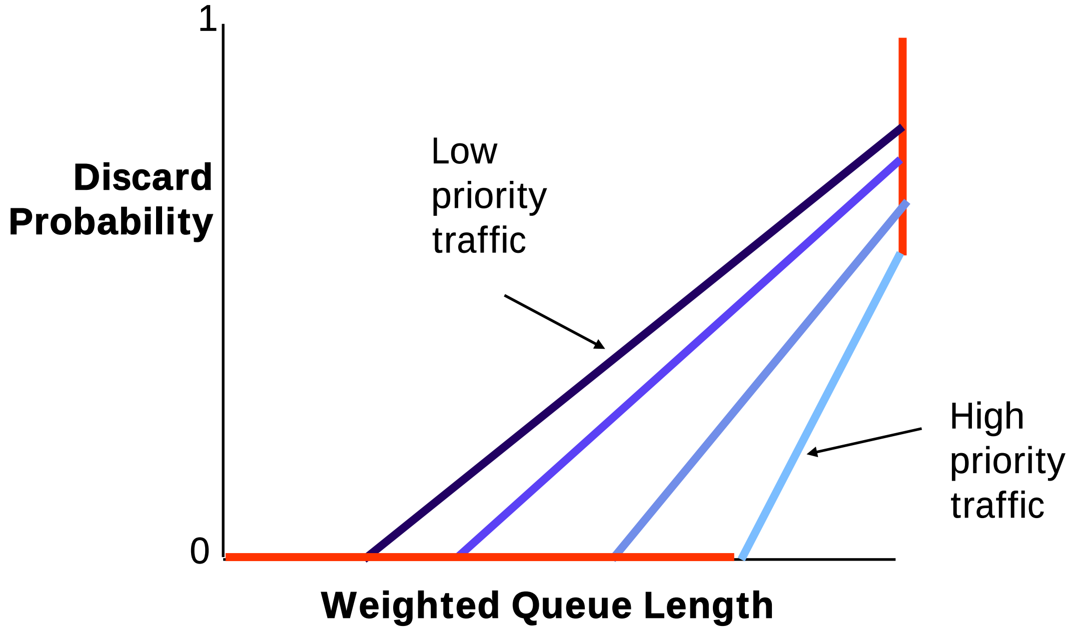
RED Algorithm

- Attempt to maintain mean queue depth
- Drop traffic at a rate proportional to mean queue depth and time since last discard



Weighted RED

- Alter RED-drop profile according to QoS indicator
  - precedence and/or drop preference



Outcomes of RED

- Increase overall efficiency of the network
  - ensure that packet loss occurs prior to tail drop
  - allowing senders to back off without need to resort to retransmit time-outs and window collapse
  - ensure that network load signaling continues under load stress conditions

Outcomes of W-RED

- High precedence and short duration TCP flows will operate without major impact
  - RED's statistical selection is biased towards large packet trains for selection of deletion
- Low precedence long held TCP flows will back off transfer rate
  - by how much depends on RED profile
- W-RED provides differentiation of TCP-based traffic profiles
  - but without deterministic level of differentiation

Pitfalls of RED

- No effect on UDP
- Packet drop uses random selection
  - Depends on host behavior for effectiveness
  - Not deterministic outcome
- Specifically dependent on
  - bulk of traffic being TCP
  - TCP using RTT-epoch packet train clustering
    - ACK spacing will reduce RED effectiveness
  - TCP responding to RED drop - but not all TCPs are created equal

Weighted RED

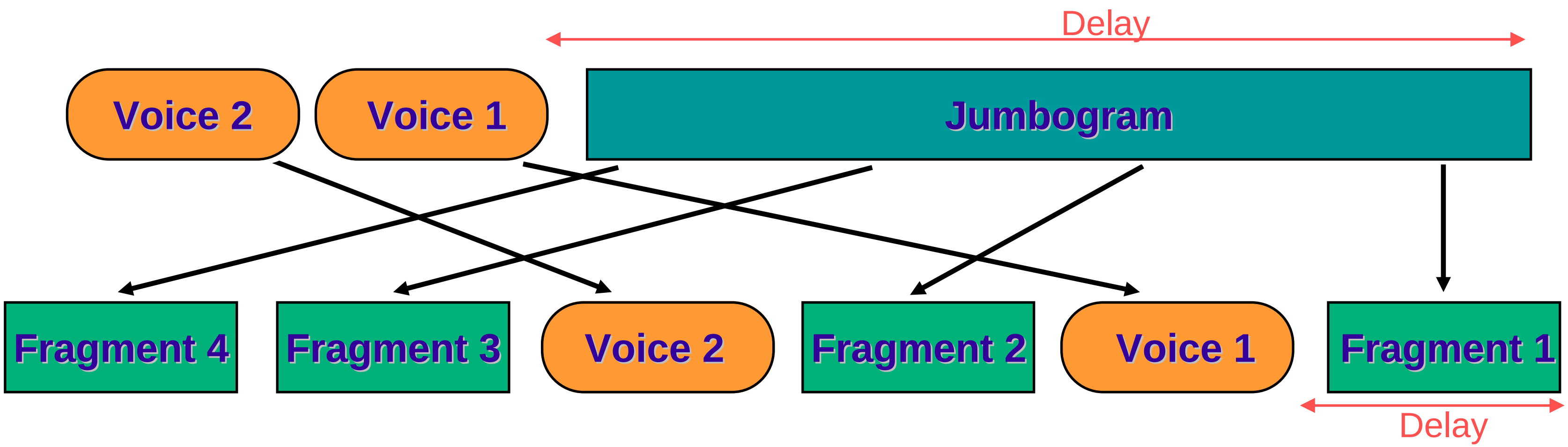
- Appropriate when
  - Any given flow has low probability of having data in queue
- Stochastic model
- Reduces turbulent inputs
- Traffic classification based on IP precedence
  - Different min_threshold values per IP precedence value

Quality of Service

Link Management

Fragmentation

PPP with fragmentation



- Fragment large packets
- Let small packets interleave with fragmented traffic

PPP with fragmentation

- You COULD define different MTU sizes per traffic QoS profile
  - lower precedence traffic has lower associated link MTU
- This is a future
  - performance impact through increased packet switching load is not well established
  - MTU discovery and subsequent alteration of QoS will cause IP fragmentation within the flow

Quality of Service

Link Management

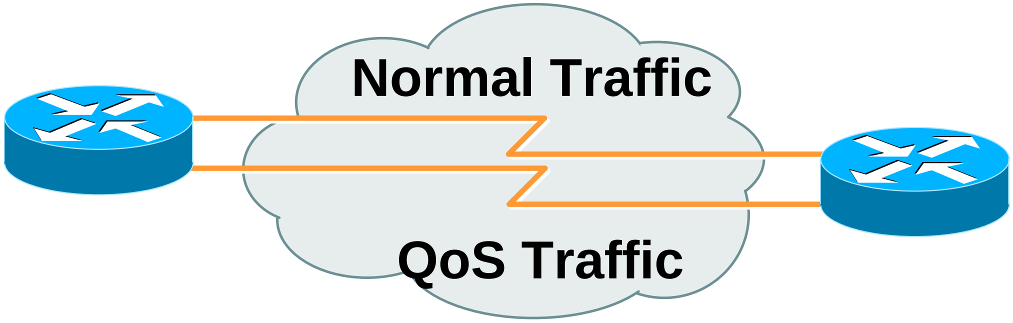
ATM Virtual Circuits

Separate VCs

- Appropriate to ATM only
- Linear behavior between VCs

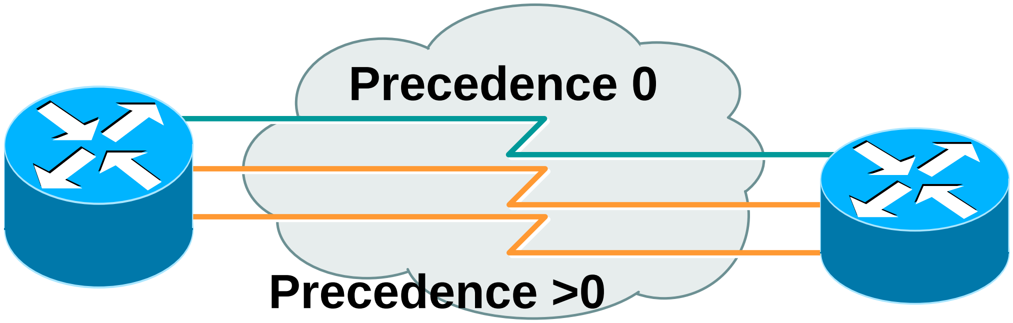
ATM with Separate VCs

- One VC per
  - Set of flows through given set of neighbors with matching QoS requirements
- Edge device routes traffic on VC with appropriate set of characteristics



ATM with per-precedence VCs

- One VC per precedence level
- Edge device routes traffic on VC with appropriate set of characteristics
- Traffic classification based on IP precedence(!)
  - High precedence traffic gets predictable service
  - Low precedence traffic can get better service from ATM network than high precedence traffic
  - Potential re-ordering within transport layer flow



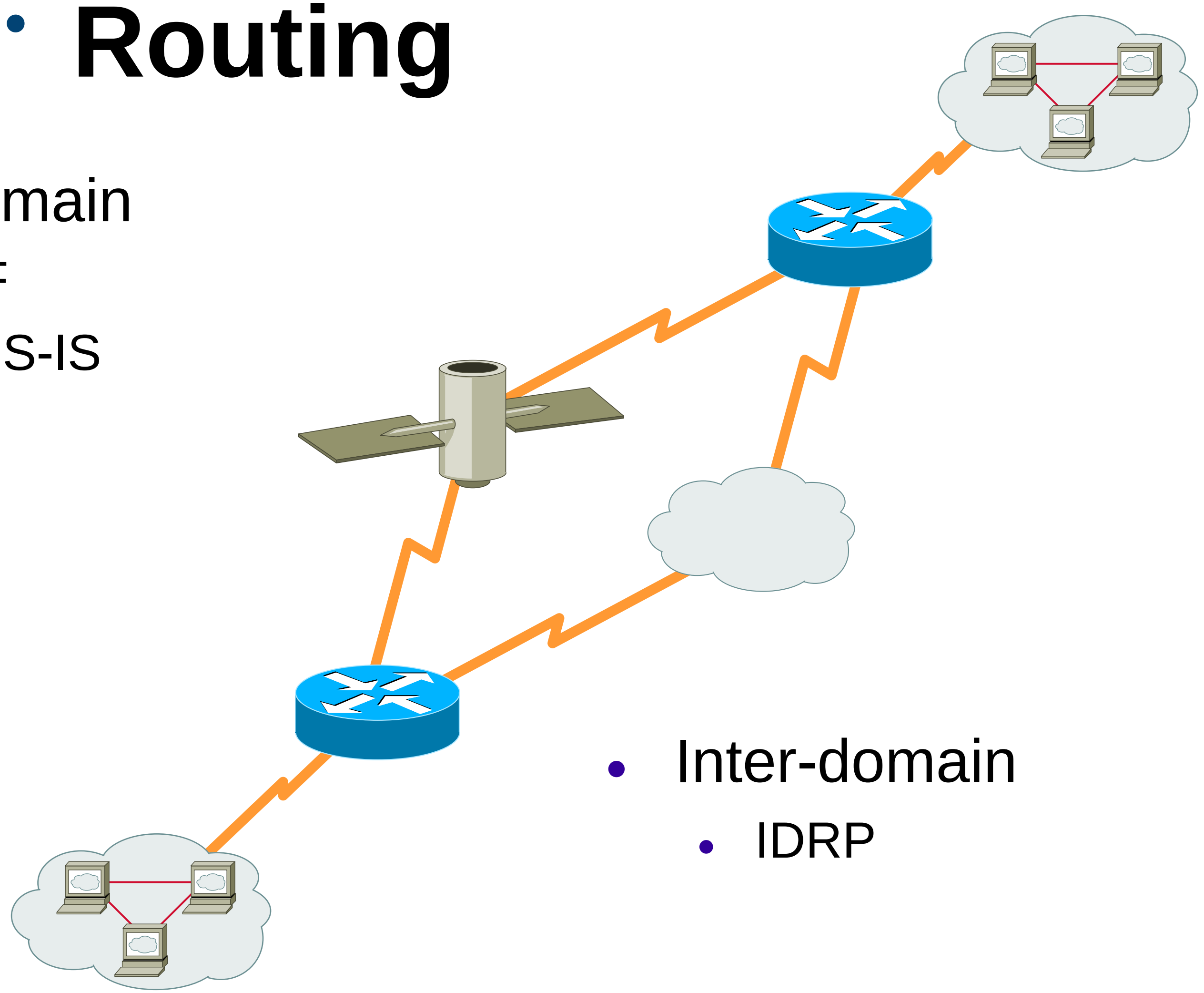
Quality of Service

Routing Management

QoS Routing

Type of Service (TOS) Routing

- Intra-domain
  - OSPF
  - Dual IS-IS



- Inter-domain
  - IDRP

Circuit Switch QoS Routing

- Sequential Alternate Routing

Sequential Alternate Routing

- Hop by hop
- Advertises
  - Available bandwidth on path
  - Hop count
- Tries to route call:
  - Successively less direct paths
  - That have enough bandwidth
- If cannot route a call
  - Tells upstream switch to try next potential path

Sequential Alternate Routing

- Observations
  - Improves the throughput when traffic load is relatively light,
  - Adversely affects the performance when traffic load is heavy.
- Harmful in a heavily utilized network,
  - Circuits tend to be routed along longer paths
  - Use more capacity.

IP QoS routing experiments

- Original ARPANET routing (1977)
- IBM SNA COS routing
- QOSPF
- Integrated PNNI

Original ARPANET routing (circa 1977)

- Ping your neighbor
  - Link metric is ping RTT
- Seek to minimize path delay
- Subject to route oscillation:
  - selected minimize delay path saturates
  - RTT rises due to queue length increase on selected path
  - alternate minimum delay path chosen

IBM SNA COS routing

- Historically, heavy manual configuration
- APPN High Performance Routing
  - Dynamic routing reduces configuration
  - Adds predictive methods to improve behavior

QOSPF

- QoS extensions to OSPF
  - Add link resource and utilization records
  - Calculate call path at each node
  - Use global state to direct this
  - Issues of simultaneity

Simulation results in QOSPF

"Thus far we have found the target environment is fully able to break any naive simulation we try."

Integrated PNNI

- Extends ATM PNNI to support IP
- Adaptive alternate path routing with crankback

PNNI algorithm combines:

- Link State (SPF)
  - Ingress node calculates full path
- Source Routing
  - Successive nodes merely accept or reject ingress node’s choice

PNNI does not address...

- Multicast routing
- Policy routing
- Alternate routing control

QoS routing constraints

- Security issues
- Policy issues
- Scaling

Flow Priorities and Preemption

- Some flows are more equal than others
  - Flow routing
  - Data forwarding
- How do we:
  - Identify these securely?
  - Bill for them?
  - Preempt existing flows in a secure fashion?

Resource Control

- Resources applied to differing QoS requirements
- Enable traffic engineering

Scale is the technical problem

- Per-flow state can be huge -- unrealistic.
- Less than per-flow routing forces unnatural engineering choices
  - All calls from A to B take same path?
  - All calls require different VCs?

Routing overheads

- State distribution
- State storage
- Route calculation

Inter-domain policy issues

- Need to handle call accounting well
  - Inter-ISP settlements...
- If route metric is path delay of a call, then a competitive service provider:
  - Possesses path data
  - Could publish the data for marketing purposes
  - Could engineer networks adversely

Now that you think you understand the problem...

Repeat the sentence using the word ‘multicast’

The QoSR plan for the moment

- Develop a framework for research
- Test protocols that appear promising

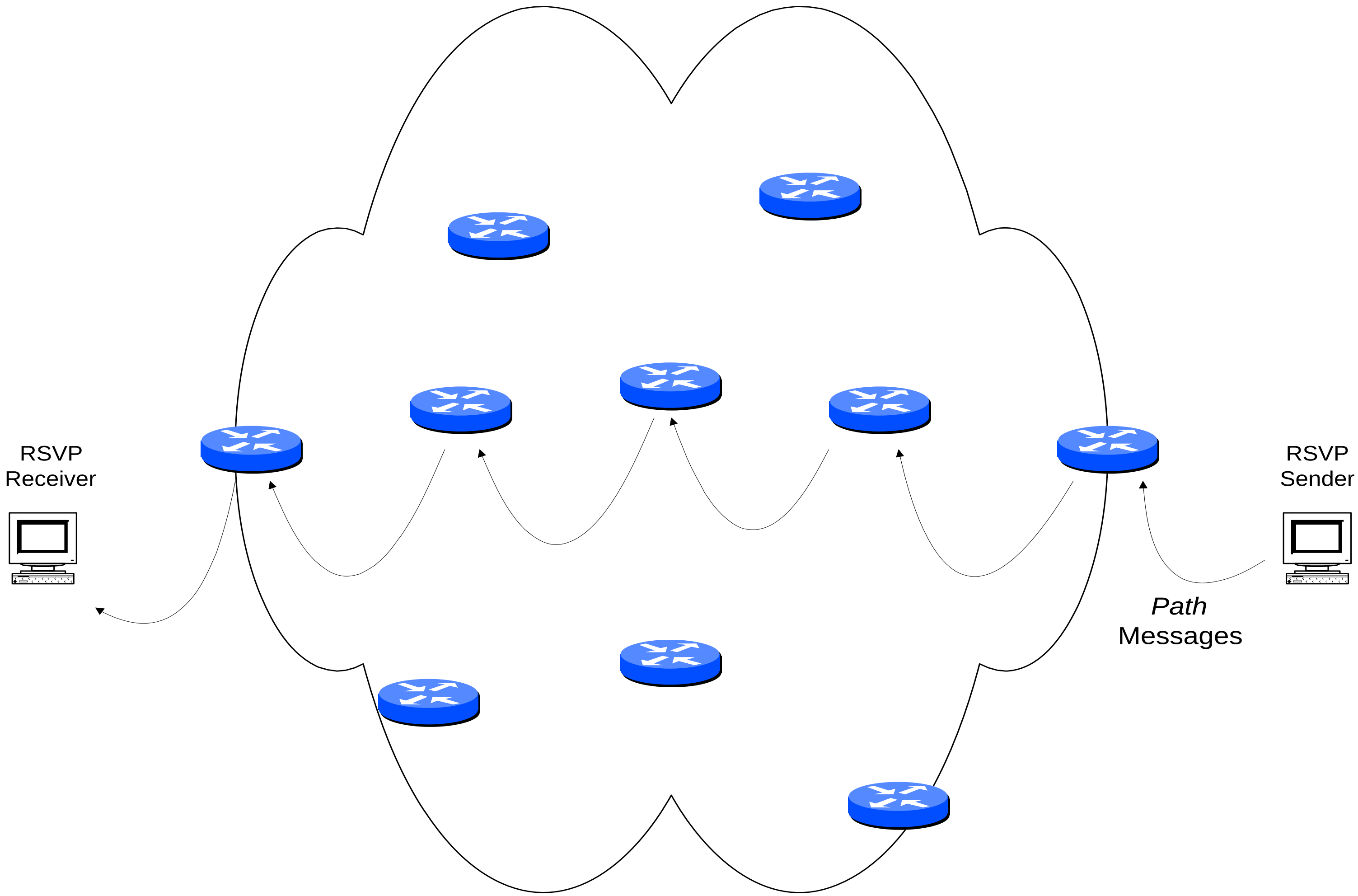
Quality of Service

RSVP

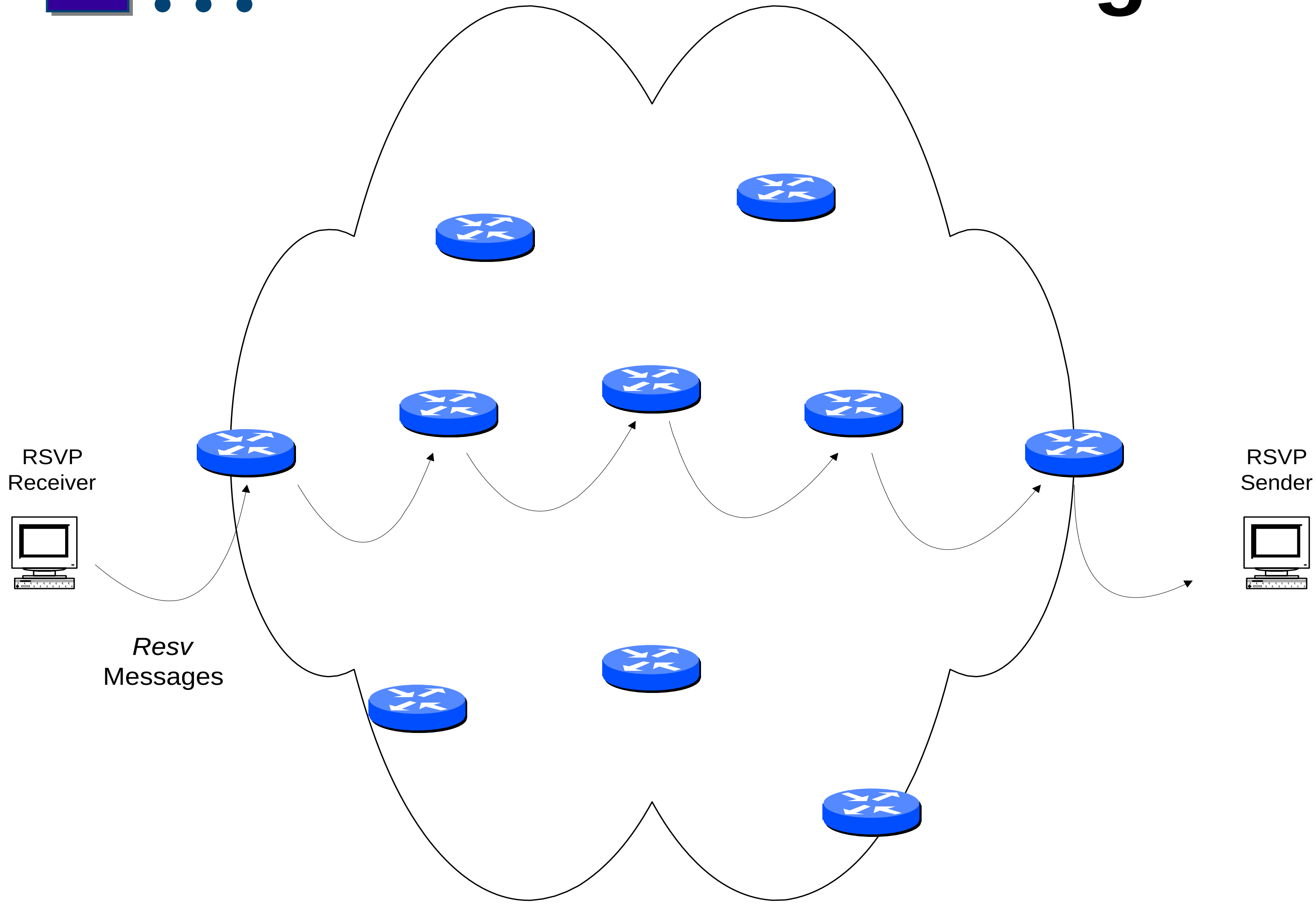
RSVP

- RFC2205, “Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification”
- RFC2208, “Resource ReSerVation Protocol (RSVP) Version 1 Applicability Statement, Some Guidelines on Deployment”
- Requires hop-by-hop, per-flow, path & reservation state
- Scaling implications are enormous in the Internet

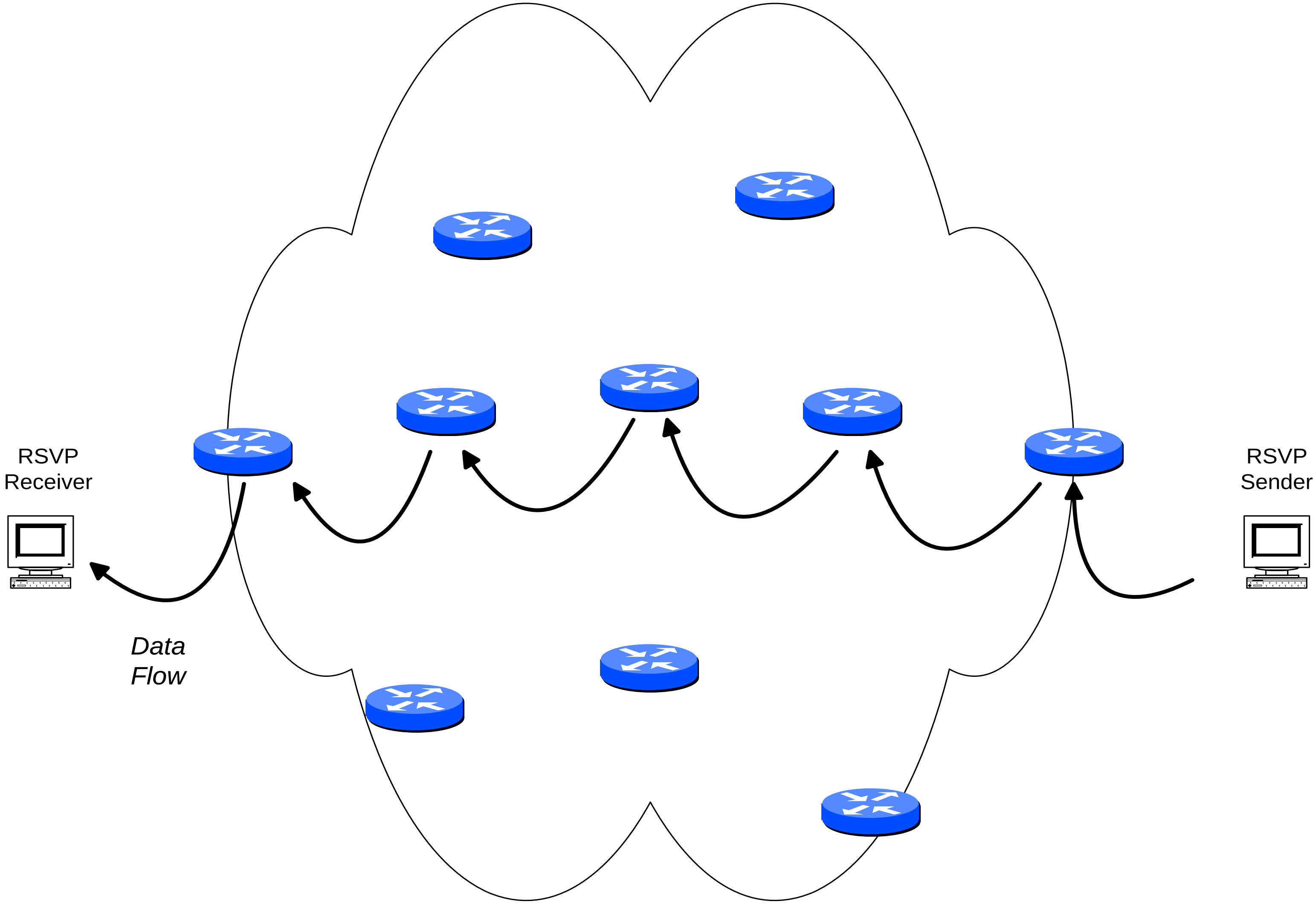
RSVP Path messages



RSVP Resv messages



RSVP Data Flow



RSVP-based QoS

- RSVP can implement service commitments:
  - Delivers a service guarantee to a preset bandwidth rate.
  - Deliver a service commitment to a controlled load profile.
  - Limit packet loss to a preset threshold.
  - Limits delay to a maximal queuing threshold.
  - Place a preset bound on jitter.
- Challenging to implement in a large network
- Relatively easy to measure success in meeting the objective

RSVP and the Internet

- RSVP requires the imposition of flow state onto network routers in order to meet the stringent requirements of a service guarantee for a flow
- Such state mechanisms do not readily scale to the size of the Internet
  - unless you want to pay the price of higher unit switching costs

RSVP Observations

- May enjoy some limited success in smaller, private networks
- May enjoy success in networks peripherally attached to global Internet
- Unrealistic as QoS tool in the Internet

Quality of Service

LAN Considerations

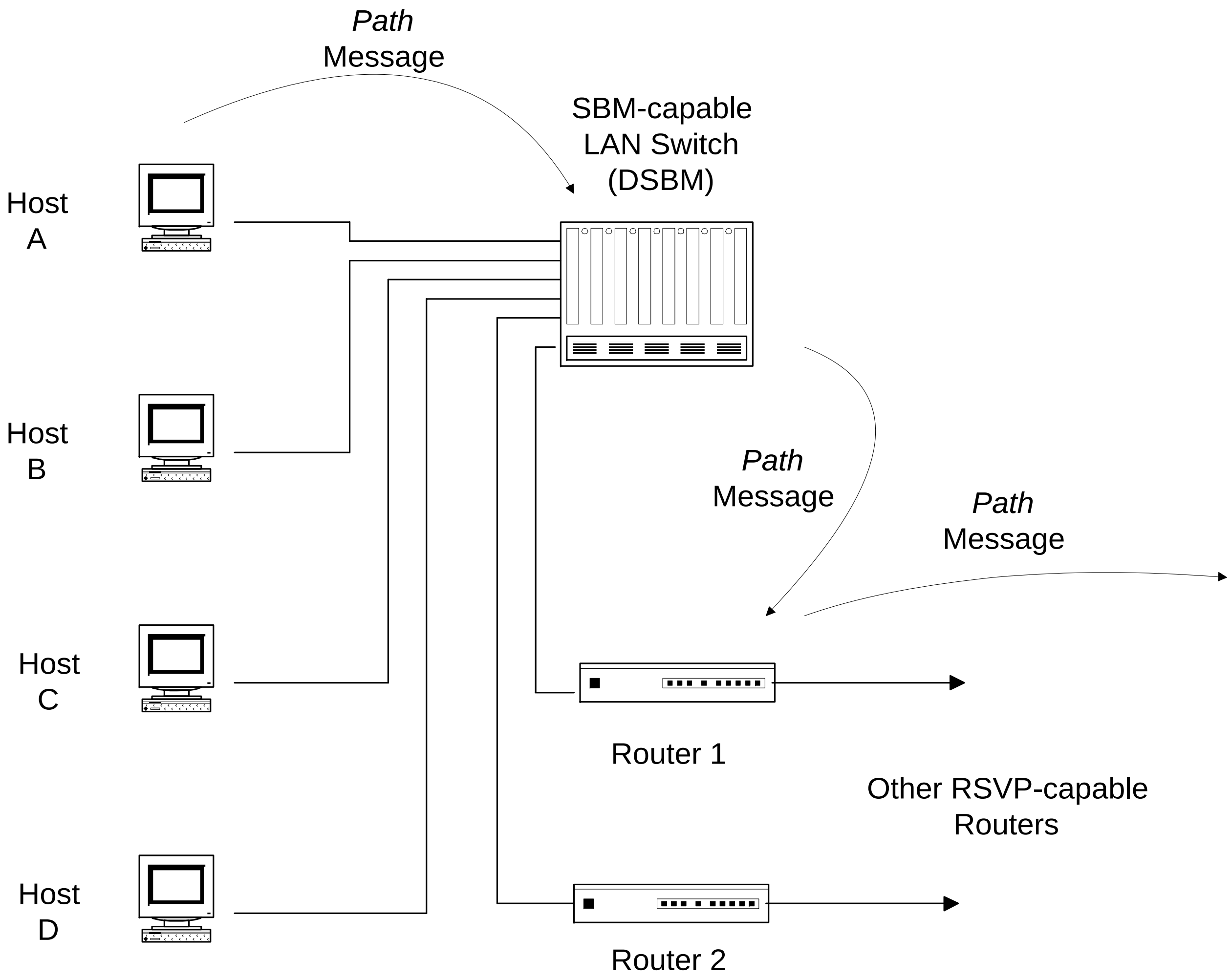
QoS and LANs

- Subnet Bandwidth Manager (SBM)
- IEEE 802.1p

Subnet Bandwidth Manager (SBM)

- IETF Internet Draft, “SBM (Subnet Bandwidth Manager): A Proposal for Admission Control over IEEE 802-style networks,” draft-ietf-issll-is802-bm-5.txt
- Integrates RSVP into traditional link-layer devices for IEEE 802 LANs
- Effectiveness is questionable without IEEE 802.1p support/integration

SBM message flow



IEEE 802.1p

- “Supplement to MAC Bridges: Traffic Class Expediting and Dynamic Multicast Filtering,” IEEE P802.1p/D6.
- Extended encapsulation (802.1Q).
- Method to define relative priority of frames (*user_priority*).
- IEEE 802.1p support in LAN switches would provide transmission servicing based on relative priority indicated in each frame (delay indication).

Quality of Service

Dial Access

QoS and Dial Access

- Most of the Internet's users connect to the Internet via dial access
- Dial access has very few built-in QoS mechanisms today

- If there is to be widespread deployment of QoS then its reasonable to expect robust and effective QoS mechanisms to be available to dial access clients

QoS and Dial Access

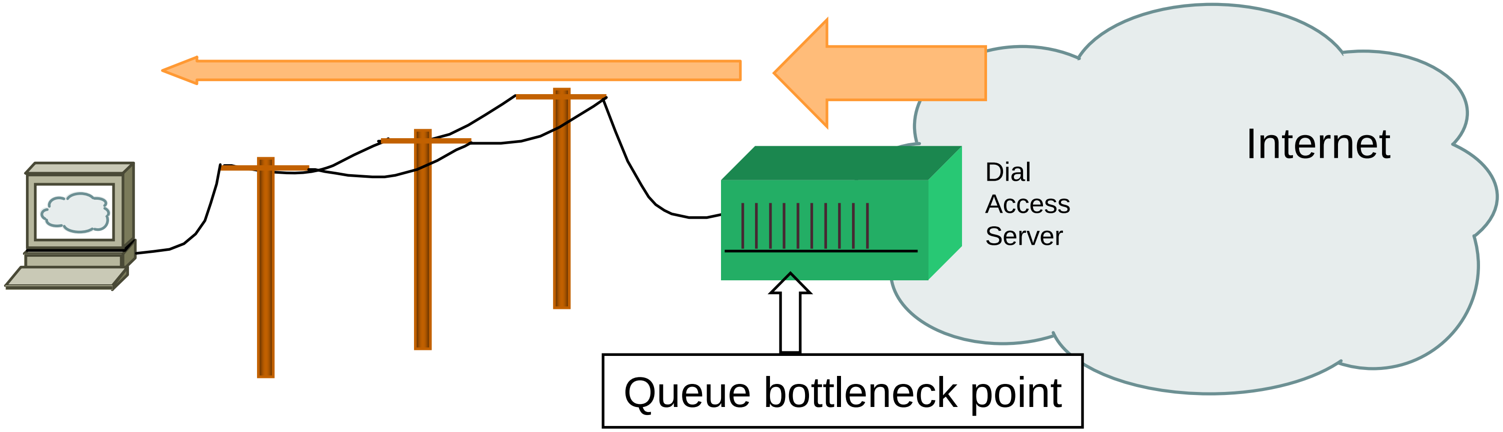
- *Service Quality First:* Port availability, no busy signals
- *Differentiation Second:* Determine methods to differentiate traffic
- *Conventional thinking:* Provide differentiation at upstream aggregation point (next hop)

QoS and Dial Access

- Port pool management
- Differentiation of port availability
  - separate port pools for each service level
    - ensure premium pool meets peak call demand levels
  - multiple logical pools using a single physical pool
    - allow incoming calls for premium access callers when total pool usage exceeds threshold level

QoS and Dial Access

- QoS with a twist:
  - A low bandwidth line requires decisions on priorities
  - QoS differentiation between simultaneous applications on the same access line
    - ie: www traffic has precedence over pop traffic
  - QoS differentiation is NOT at the host
  - QoS differentiation is at the dial access server

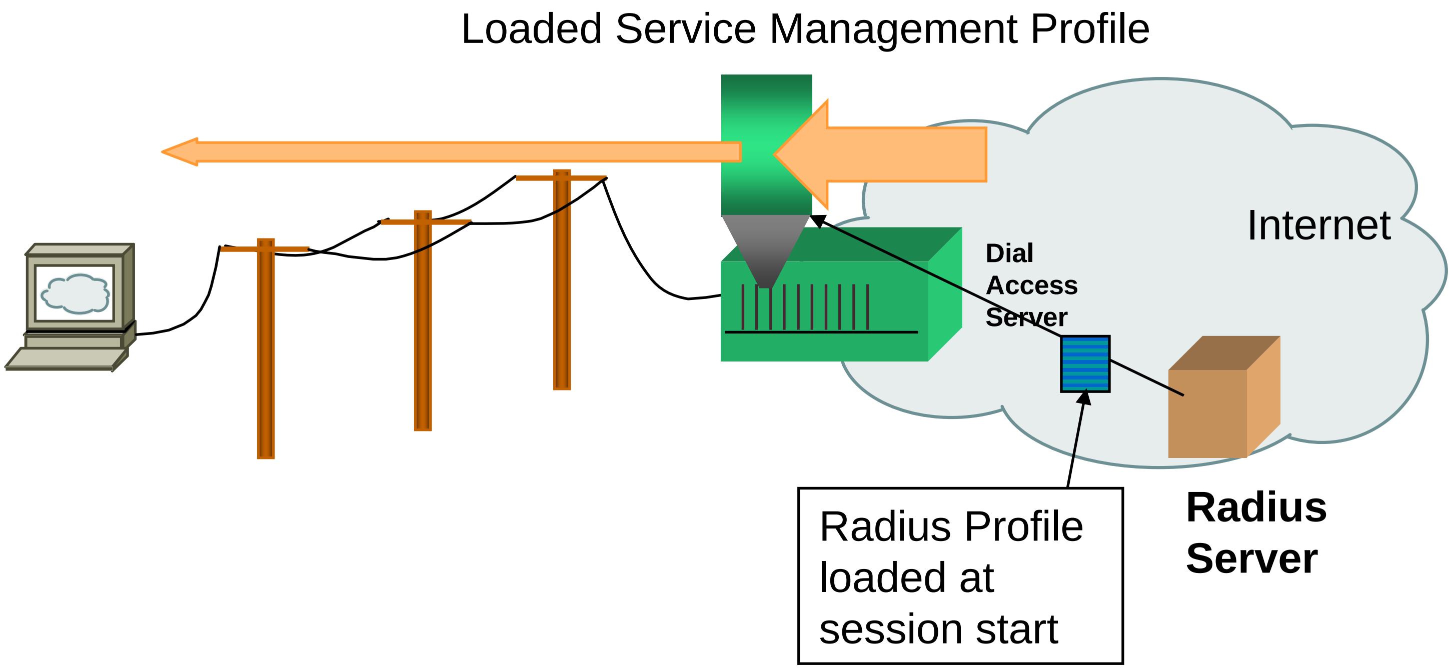


QoS and Dial Access

- The problem here is how to load service management rules into the dial access server to implement the user's desired service profile
- Radius profiles
  - per-user profiles are loaded in the dial access server as part of session initiation
  - use radius extensions to load service profile
  - no changes required to host environment

QoS and Dial Access

- Radius service profile



Quality of Service

Measuring QoS

QoS Measurement Tools

- QoS measurement tools available today:

QoS Measurement Tools

- We **can** instrument tools to measure network
  - delay
  - jitter
  - bandwidth
  - reliability

  on a specific path, at a specific times
- But we cannot measure network-wide QoS
  - its not a concept which translates into an artifact which is directly measureable

QoS Measurement Tools

- **SNMP** monitoring of routers
- real time monitoring of 'network component health' by measuring for each link:
  - link occupancy
  - packet throughput
  - mean queue depth
  - queue drop levels
- Use these metrics to create a **link congestion metric**

QoS Measurement Tools

- Host-based probes to measure path state
  - *ping*
    - measures end-to-end delay & jitter
  - *bing*
    - provides per-hop total bandwidth estimate by differential timing across a single hop
  - *traceroute*
    - delay, and reliability measurement through path trace
  - *treno*
    - measures available bandwidth through TCP reno simulation with ping packets

QoS Measurement

- SNMP and host probes
  - What the user wants to measure
    - the difference in performance between QoS differentiated and ‘normal’ service transactions
  - What the tools provide
    - a view of the performance of the components of the network, not a view of the performance of a network transaction

Host Measurement of QoS

- Measure the TCP transaction at the sender
  - stability of RTT estimate
  - stability of congestion point (available bandwidth)
  - incidence of tail drop congestion
  - incidence of timeout
- Sustainable TCP transfer rate

Host Measurement of QoS

- Measure the TCP transaction at the receiver
  - incidence of single packet drop
  - incidence of tail drop congestion
  - duplicate packets
  - out of order packet
- Sustainable TCP transfer rate

Host Measurement of QoS

- Measure the UDP service at the reciever
  - measure signal distortion components
    - loss, jitter, delay, peak received rate

QoS Measurement

- Internet performance metrics are still immature
- We have
  - tools to measure individual artifacts of performance
  - Tools to measure individual session performance

QoS Measurements

- We **don't** have
  - tools to measure *quality potential*
    - "As a client, if I were to initiate a session with elevated priority how much faster would the resultant transaction be?"
  - tools to measure *quality delivery*
    - "As a provider, am I providing sufficient resources to provide discernable differentiation for elevated quality services?"

Quality of Service

Host Considerations

QoS in the network is fine, BUT...

- performance of an application is dependant on
  - the state of the network
  - the sender and the receiver
- poor performance is often the outcome of poor or outdated protocol stacks

Performance Issues

- Major benefits can be gained by using protocol stacks which support:
  - large buffers and window scaling options
  - selective acknowledgement (SACK)
  - correct RTT estimate maintenance
  - correct operation of window management algorithms
  - MTU discovery
  - initial cwnd value of 4
- and use hosts with
  - enough memory and CPU to drive the protocol

Network or Host

- QoS in the network is not always the right answer to the question of poor performance.
- Often the problem is the box behind your screen!

Quality of Service

Marketing QoS

Marketing

- What is being marketed?
  - current base is variable best effort
  - *better* than best effort?
    - PREMIUM SERVICE
  - *worse* than best effort ?
    - BUDGET SERVICE
  - constant effort ?
    - DEDICATED SERVICE

Marketing

- Pricing and QoS
  - sender pays for premium / discount service
  - service level set at packet ingress based on source admission policy
    - at ingress?
    - at egress ?
  - receiver pays for premium / discount service
  - service level set at packet ingress based on destination admission policy
    - at egress

Marketing QoS is not easy...

- Premium / Discount services are relative to best effort Base
- Base level service quality varies on current load and path
- Differentiated service levels difficult to quantify to customer
- How are Service Level Agreements phrased for differentiated service environments?

Marketing

- High variability of base level service is an impediment to marketing QoS
- Marketing QoS will also require
  - service level agreements
  - dissemination of robust measurement tools able to measure differential quality
  - accurate expectation setting

Quality of Service

Summary

Current QoS Mechanisms

- Rate Control
  - token bucket
  - leaky bucket
    - admission mechanisms
- Queuing control
  - weighted Fair Queuing
  - weighted RED
    - internal resource management mechanisms

Current QoS Mechanisms

- Link control
  - Parallel Virtual Circuits
  - MTU management
    - Data Link layer differentiation
- RSVP
  - guaranteed load service
  - controlled load service
    - limited environment of deployment

Current QoS Mechanisms

- Routing
  - QoS Routing
    - experimentation proceeds!

QoS Implementation Considerations

- Complexity:
  - If your support staff can’t figure it out, it is arguably self-defeating
- Delicate balance between quality of network design & architecture and QoS differentiation mechanisms

Yet to be Resolved

- Only long-held adaptive flows are highly susceptible to network layer shaping
  - Symmetric handling of TCP QoS service requests
  - Short held flows (WWW transactions) are not very susceptible to network layer shaping
- UDP flow management
  - Unicast flow model (ingress filter or sender moderation?)
  - Multicast flow model (multi-layering of the signal?)
- Inter-Provider semantics & agreements for differentiated services

Unanswered Questions

- How does the provider measure QoS?
- How does the customer measure QoS?
- How do you tariff, account, and bill for QoS?
- How will QoS work in a heterogeneous Internet?
  - QoS across transit administrative domains which may not participate or use different QoS mechanisms?

Summary

- There are no absolute guarantees in the Internet. Sorry.

Summary

- There is no magic QoS bullet. Sorry.

Summary

- There is possibly a “middle ground” somewhere between traditional single level best effort and guaranteed customized services.

References

- *Quality of Service: Delivering QoS in the Internet and the Corporate Network*
  http://www.wiley.com/compbooks/ferguson/
- *Differential Services in the Internet*
  http://diffserv.lcs.mit.edu/

Questions?

Thank you.

Geoff Huston
Technology Manager
Telstra Internet
gih@telstra.net