QoS in IP networks

Computer Science Department
University of Crete
HY536 - Network Technology Lab II
2000 - 2001

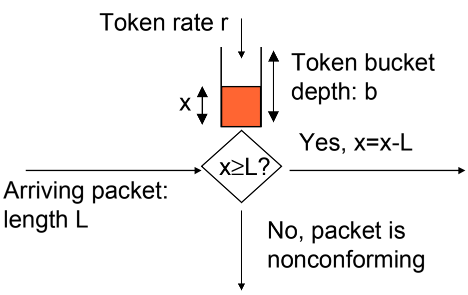
IETF Integrated Services (IntServ)

- Connection-oriented solution (end-to-end)
- QoS guarantees on a per flow basis
- Intermediate router keep per flow state
- Building blocks:
  - resource reservation protocol (RSVP) for end-to-end signaling
  - admission control
  - policing: check if traffic conforms to profile
  - shaping: modify traffic timing so that it conforms to profile
  - classification: identify packets that are to receive certain level of service
  - scheduling: isolate flows and support minimum bandwidth

Integrated Services Architecture

- **Guaranteed Service:**
  - deterministic delay guarantee (mathematically provable); zero packer loss
  - token bucket used to specify traffic
  - specification of requested service
- **Controlled-Load Service:**
  - network provides service close to that provided by a best-effort network under lightly loaded conditions
  - token bucket used to specify traffic
- **Best-Effort Service:**
  - no guarantees

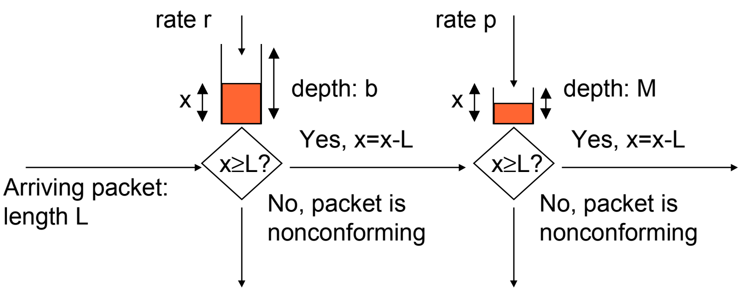
Token bucket



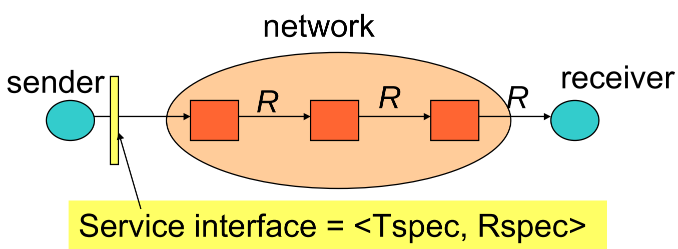
- Equivalent to leaky bucket
- Amount of data over time T: $D(T) \leq rT+b$

Full token bucket specification

- Three additional parameters:
  - minimum policed unit m: policer required to remove at least m tokens for each conforming packet
  - maximum packet size M: largest permissible packet size
  - peak rate p
- Amount of data over time T: $D(T) \leq \min\{pT+M, rT+b\}$

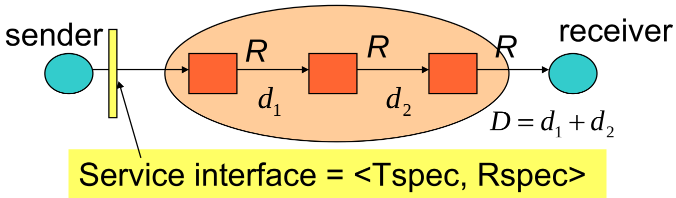


Guaranteed QoS service class



- Traffic Specification: Tspec = (r, b), p, M, m
- Service request specification: Rspec = (R,S) minimum reserved capacity
  - S is a slack term representing the difference between the required delay and the maximum delay using reservation R
- controls maximum delay, not minimum, average, or jitter

Maximum delay for guaranteed services



- From Tspec & Rspec can compute maximum delay
  - Example: fluid model, m = 0, M = ∞

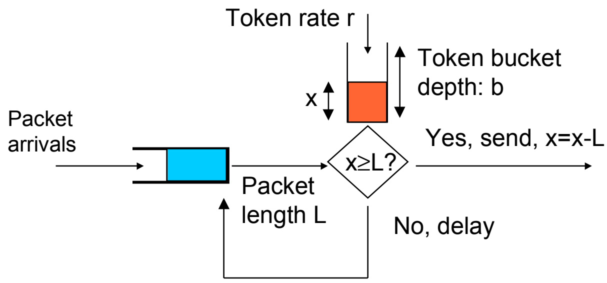
$$
\begin{aligned}
\text{end-to-end delay} &= \frac{b}{p-r}\,\frac{p-R}{R} + D \quad \text{if } p > R \\
&= D \quad \text{if } p \le R
\end{aligned}
$$

- Parameter selection:
  - Given Tspec, D, Dmax the application finds R (= Rspec)
  - Given Tspec & Rspec network chooses the buffer required for zero cell loss

Policing and shaping for guaranteed services

- Policing performed at ingress of network
  - non-conforming packets treated as best-effort
  - possibility of out of order delivery (bad, e.g. for real-time)
- Reshaping done at intermediate point of the network
  - may be necessary due to distortions as traffic flows through network

Token bucket shaper



Controlled-load service class

- Intended to support applications highly sensitive to overloaded conditions
- Service provided tightly approximates service of best-effort networks under unloaded conditions
  - very high percentage of transmitted packets will be successfully delivered
  - transit delay experienced by a very high percentage of delivered packets will not greatly exceed minimum transmit delay
- Uses only Tspec = (r, b), p, M, m and not Rspec

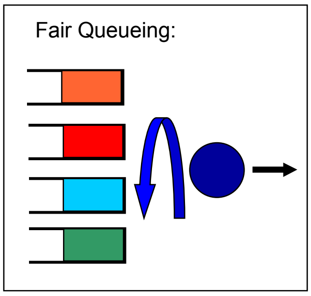
Scheduling

Objectives:

- give different flows a different bandwidth share
- support minimum bandwidth guarantees
- isolation: one flow cannot monopolize whole resource
- implementation, admission control decisions, etc

Schemes:

- FIFO
  - no isolation
- Priority Queueing
  - high priority can starve lower priority
- Fair Queueing/Weighted Fair Queueing
  - each flow gets share of bandwidth
  - isolation of flows
- Class Based Queueing
  - proportional bandwidth sharing among classes



IntServ and ATM: similarities

- Both require signaling
- Both operate on per flow basis
- Both use admission control

IntServ and ATM: differences

- ATM: hard state
- IntServ: soft state
  - need to periodically refresh reservation
  - refresh request can be denied
  - user allowed to change reservation
- ATM more “predictable” network
- ATM: connection-oriented nature provides confirmation of setup or denial
- IntServ: no sender control
- ATM QoS negotiable
- IntServ: Guaranteed service determined from Tspec,Rspec; not negotiable for controlled load

IntServ and ATM: issues

- Complexity in routers: packet classification & scheduling
- Scalability in core since both operate on per-flow basis
- Ease of deployment
- Need concept of “virtual paths” or aggregated flow groups in core

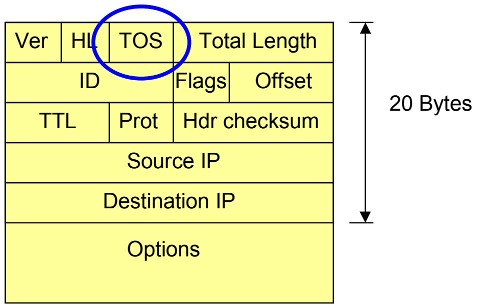
IETF Differentiated Services

- Goal: offer differing levels of performance (Quality of service - QoS) to different users
  - improve revenues (premium pricing)
  - competitive differentiation
- Key concepts:
  - scalability
  - simple model:
    - traffic entering network is classified into a small number of classes
    - a class (“behavior aggregate”) is characterized by a tag (“DS Code Point”)
    - a router services packets according to the tags
    - QoS per class (aggregate traffic), not per individual flow
  - keep forwarding path simple to allow easy and early deployment; push complexity to network edge

IETF Differentiated Services (cont.)

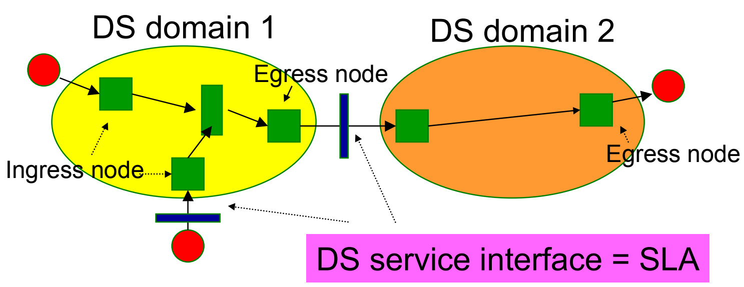
- Key concepts:
  - avoid “strong” assumptions on traffic types
  - marking based on static/long term “Service Level Agreements” (SLAs); avoids signaling
  - don’t develop/specify services, but rather
  - standardize “Per Hop Behaviors” (PHBs); but leave some DS Code Point patterns for experimental and local use
  - use PHBs to construct services
  - ability to provide services depends on ability to manage and configure routers in a coordinated manner

IETF Differentiated Services (cont)



- Use TOS (renamed to DS) 6 bit field to sort packets into classes and treat them differently

IETF DS Architecture



- DS region: one or more DS domains
- Scope of service: one-directional traffic, point-to-multipoint,
- across domains
- QoS: quantitative, qualitative
- Dynamic and static SLAs

IETF DS Architecture (cont.)

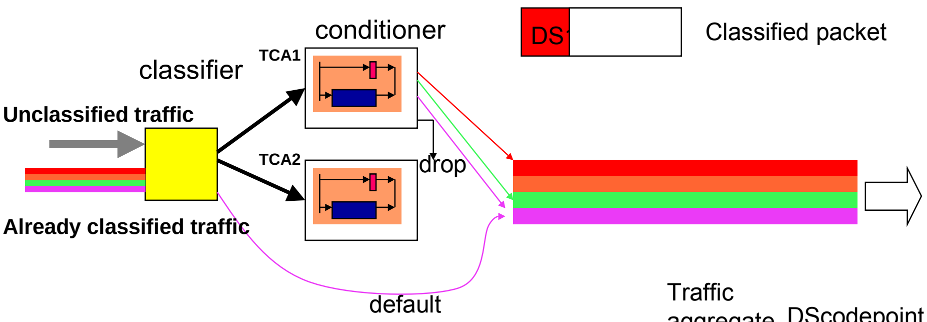
- Service Level Agreement (SLA):
  - Traffic Conditioning Agreement (TCA)
    - QoS (qualitative or quantitative)
    - traffic profile
    - disposition of excess traffic
    - marking services
    - shaping services
  - Availability, reliability
  - Routing
  - Encryption, authentication, monitoring, auditing
  - Other responsibilities
  - Pricing and billing mechanisms

DiffServ components

- Edge node algorithms
  - classification
  - metering
  - policing (dropping)
  - shaping
  - marking

  Conditioners
- Router algorithms
  - packet discard algorithms
  - priority, scheduling

Operations at boundary nodes



	Traffic aggregate	DScodepoint
	1	DS1
	2	DS2
	3	DS3
Best-effort	4	DS4

Classification: based on
- DS codepoint (Behavior Aggregate classifier), or
- combination of src/dst address, protocol, port numbers, etc

Conditioning: enforces TCA
- discard, shape, mark, meter

Meters

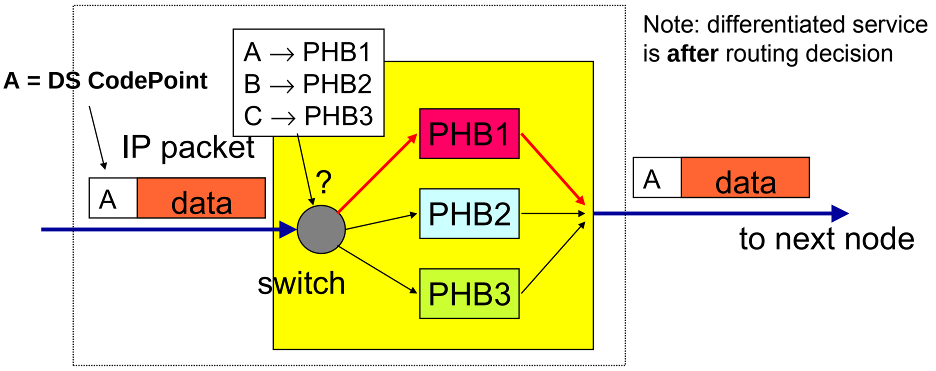
Meters monitor packet arrival and determine the level of conformance of each packet to pre-established profile

Possible meters:

- two parameter token bucket
- multistage token bucket
- average rate meter
- exponential weighted moving average meter

Router operation

Router model: set of output links, each link offers a set of services
Link service = Per Hop Behavior (PHB)

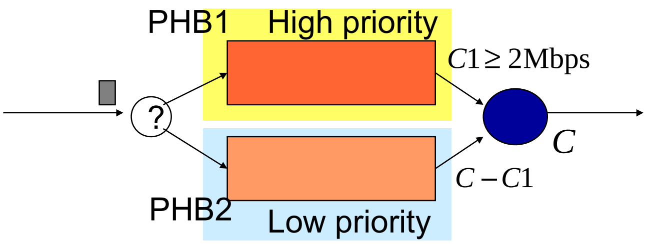


Output link model

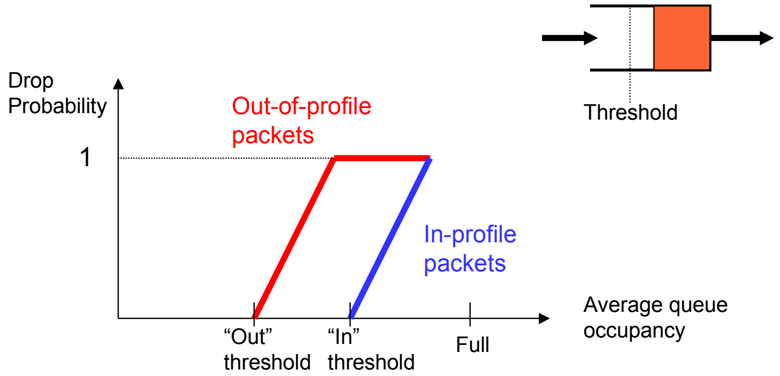
All nodes of a DS domain must have same mapping DSCP → PHB

Priority queueing

Priority queue with bandwidth reservation



Discard policy: RIO



- RIO: Random Early Detection (RED) with In and Out

Expedited Forwarding PHB

- Provides guaranteed service rate with negligible delay/jitter
- "virtual leased line"
  - but doesn't waste bandwidth when line is not used
- Guaranteed minimum service rate
- Policing: arrival rate < minimum service rate
- Not affected by other PHBs
  - highest priority, in case of priority scheduling
  - expected to use a small portion of network bandwidth

Comparison Expedited Forwarding with ATM CBR

Both services: traffic will be shaped and peak rate will be guaranteed

ATM CBR:

- VC needs to be set-up (signaling)
- “hard” admission control

Expedited forwarding:

- no signaling
- long term provisioning
- admission control performed off-line

Assured Forwarding PHB group

- Provides different levels of forwarding performance
- PHB group (not single forwarding behavior)
- Four classes
  - each allocated, e.g. using class-based queueing, a certain amount of resources (buffer, capacity)
- Three drop preferences per class
- No reordering of packet belonging to same microflow, even if some are out-of-profile

Comparison of Assured Forwarding with ATM

Similar to nrt VBR/ABR

nrt VBR/ABR:

- VC needs to be set-up (signaling)
- VBR: admission control
- ABR: supports flow control (but flow control performed at TCP)

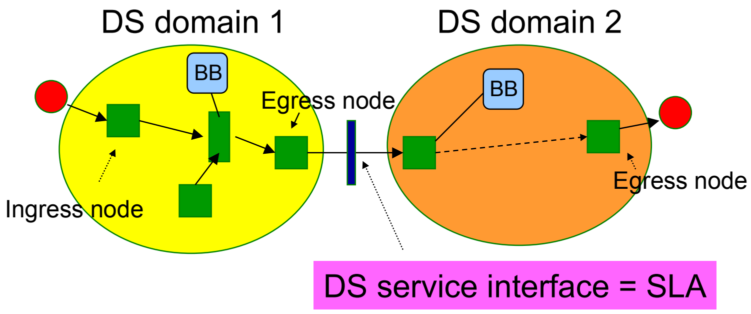
Assured Forwarding:

- no signaling
- long term provisioning
- admission control performed off-line

DiffServ issues

- Guarantees provided for aggregate traffic
  - good for ISP, but what about end-users
- Unidirectional (no receiver control)
- Guarantees can be affected if packets follow different paths
- End-to-end service depends on PHBs at every hop
  - end-to-end $\neq$ Σ per-hop; e.g. weighted guarantees at every hop
- Designed for long-term SLAs
  - both network topology and traffic are highly dynamic
  - dynamic SLAs possible (time-scales?)
- Network management needed for resource control and for guaranteeing QoS

Bandwidth Brokers



- Bandwidth Broker (BB):
  - user requests service from bandwidth broker
  - decides which users can request special service (policy)
  - tells edge routers what to tag (configuration)
  - decides if SLAs can be accepted (resource/admission control)

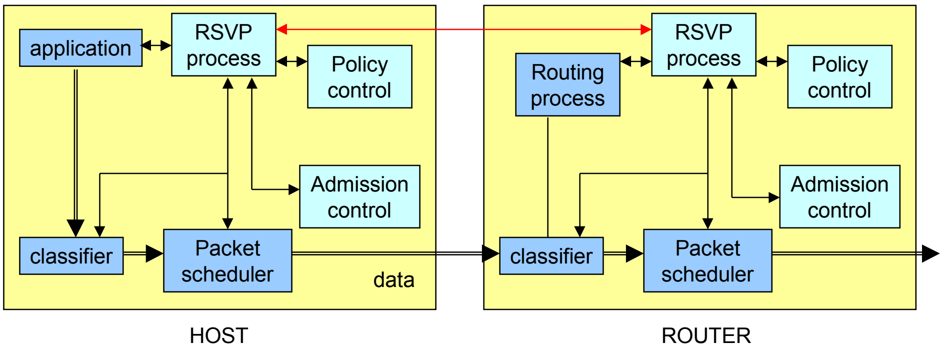
RSVP and ATM signaling

Resource Reservation Protocol (RSVP) and Q.2931 for ATM networks are out-of-band signaling protocols

ATM (Q.2931)
- sender initiated
- bidirectional
- reliable means to inform regarding request status

RSVP
- receiver initiated
- unidirectional
- no reliable means to inform regarding request status

RSVP operation with IntServ

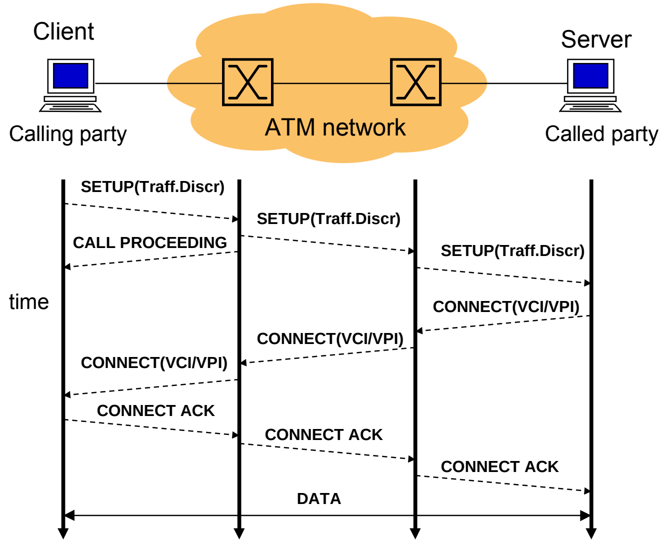


- Works in co-operation with routing protocol (needs QoS support)
- RSVP does not understand contents of message it carries

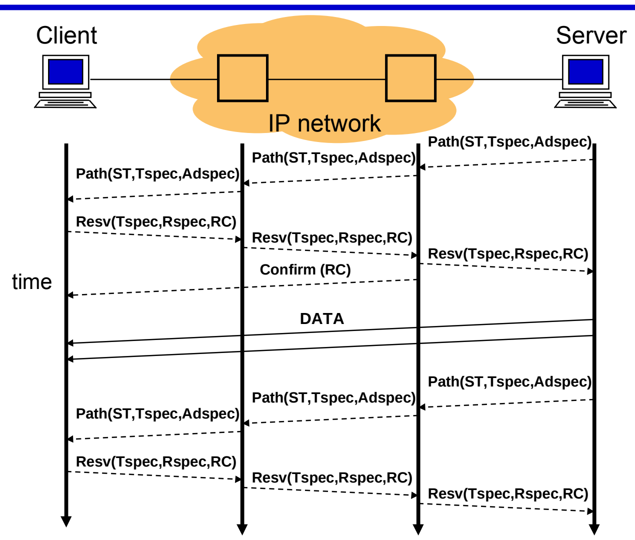
RSVP objects

- Flowspec:
  - Rspec $\rightarrow$ requested service (QoS)
  - Tspec $\rightarrow$ traffic description
  - used to set parameters of packet scheduler
  - opaque to RSVP
- Filterspec:identifies a flow
  - based on IP sender/receive address, port, etc.
  - used to set parameters of packet classifier

ATM signaling



RSVP



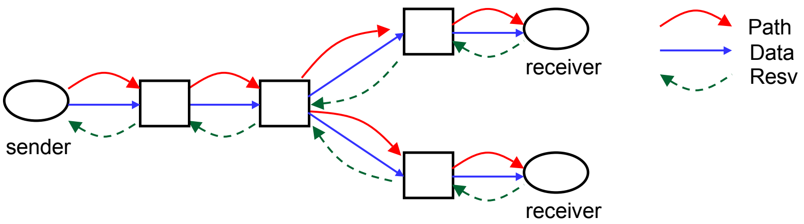
- ST: Sender Template (sender IP address, port #)
- RC: Resv Confirm (sent by merge point or sender); can be lost

RSVP: use of Adspec

- Advertisement specification (Adspec): contained in Path message, includes Ctot, Dtot
- Used by receiver to compute maximum end-to-end delay of guaranteed service

$$\begin{aligned} \text{end-to-end delay} &= \frac{b-M}{p-r}\frac{p-R}{R}+\frac{M+Ctot}{R}+Dtot \quad &\text{if } r \le R < p \\ &= \frac{M+Ctot}{R}+Dtot \quad &\text{if } r \le p \le R \end{aligned}$$

RSVP operation



- Each router finds previous hop from Path message
- Resv message follows opposite route of Path messages
- Upon receipt of Resv message router performs
  - admission and policy check
  - if admitted, set classifier/scheduler & propagate upstream (upstream reservation may be different due to merging)
  - if not admitted, send ResvErr
- Resv propagated upstream until an existing reservation greater than or equal to the requested is encountered

RSVP reservation styles

- Users of multicast applications receiver flows from different senders
- A separate reservation request is required for each flow
- RSVP supports a flexible way to reserve resources for flows from different senders using reservation styles

reservation

Sender selection		Distinct	Shared
	Explicit	Fixed-filter (FF) style	Shared-explicit (SE) style
	Wildcard	-	Wildcard-Filter (WF) style

RSVP basic properties

- Assumes existence of multicast/QoS routing and admission control
- Accommodate receiver with heterogeneous requirements
- Adapt to changes of multicast group membership
- Exploit requirements of different receivers to use network resources efficiently
- Reserve more resources only when needed
- Allow receivers to dynamically connect to sessions
- Adapt to changes of routes
- Soft state: need to periodically refresh reservation