Better-than-best-effort: QoS, Int-serv, Diff-serv, RSVP, RTP

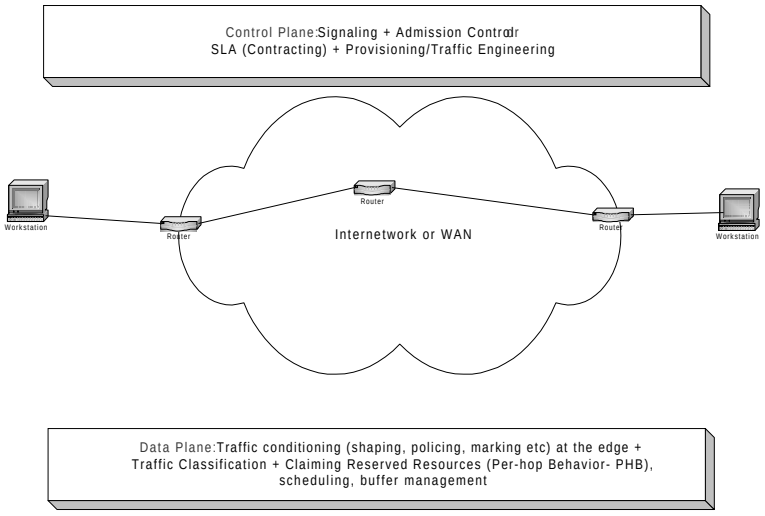
Shivkumar Kalyanaraman
Rensselaer Polytechnic Institute
shivkuma@ecse.rpi.edu
http://www.ecse.rpi.edu/Homepages/shivkuma

- QoS building blocks
- ATM QoS architecture
- Why better-than-best-effort Internet ?
- Support for multimedia apps: RTP, H.323, Integrated Services(int-serv), RSVP.
- Scalable differentiated services for ISPs: diff-serv
- Missing pieces: QoS routing, traffic engineering, policy management, pricing models

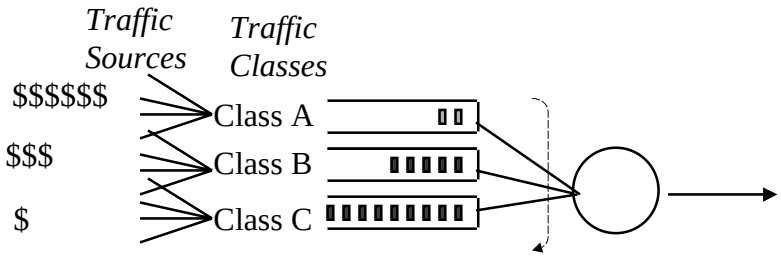
QoS building blocks

- QoS => set aside resources for premium services
- QoS components:
  - a) What kind of premium services ? (***Service/SLA design***)
  - b) How much resources? (***admission control/provisioning***)
  - c) How to ensure network resource utilization, do load balancing, flexibly manage traffic aggregates and paths ? (***QoS routing, traffic engineering***)
  - d) How to actually set aside these resources in a distributed manner ? (***signaling, provisioning, policy***)
  - e) How to deliver the service when the traffic actually comes in ? (***traffic shaping, classification, scheduling***)
  - f) How to monitor quality, account and price these services? (***Network management, Accounting, Billing, Pricing***)

QoS big picture: Control/Data planes



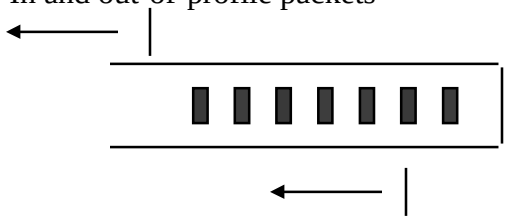
Eg. Mechanisms: Queuing/Scheduling



- Use a few bits to indicate which queue (class) a packet goes into (also branded as CoS)
- High $$ users get into high priority queues, which are in turn less populated => lower delay and near-zero likelihood of packet drop

Eg. Mechanisms (contd): priority drop

Drop In and out-of-profile packets



Drop only out-of-profile packets

- Enhance buffer management to preferentially drop red packets when a low threshold is crossed

ATM QoS framework

- ***Services***: CBR, rt-VBR, nrt-VBR, ABR, UBR
- ***QoS Routing and Signaling***:
  - PNNI, ATM signaling with VCs/VPs
- ***Traffic management:***
  - QoS parameter design, traffic conditioners, feedback control
  - Standard end system and switch behavior for each of the services
- ***Critique:*** No support for qualitative, provider-defined services, limited pt-to-mpt support

ATM Traffic Classes

- **CBR,VBR** for voice, video:"higher priority"
- **ABR, GFR, UBR** for data: uses "left over capacity"
- **ABR properties:** low latency, high throughput, fairness among contending sources, and low cell loss.
- **UBR properties:** No guarantees. Happy-go-lucky.
- **GFR properties:**
  - Minimum rate provided through simple signaling and buffer management.
  - Intermediate to ABR and UBR - similar to frame relay.

Internet real-time support model

- Initially assume that the net offers no real-time support and engineer transport protocols (RTP) and middleware which can enable adaptive real-time applications
- On the longer term, build QoS mechanisms: control-plane and data-plane
- Flexibility to leverage the Internet connectionless model, allow for future multicast capability, accommodate ISP's desire to "provision/engineer" networks, and design their own services

RTP

- RTP is the standard protocol for the transport of real-time data, including audio and video.
- RTP follows the application level framing (ALF) philosophy.
  - RTP specifies common app functions.
  - It is intended to be tailored through modifications and/or additions to the headers (spec'd in companion docs)
- RTP consists of a data and a control part. The latter is called RTCP.
- The data part of RTP is a thin protocol.

RTCP

- RTCP provides support for real-time conferencing of groups of any size within an internet.
  - Eg: source identification and support for gateways like audio and video bridges as well as multicast-to-unicast translators.
  - It offers quality-of-service feedback from receivers to the multicast group & synchronization support for media streams.

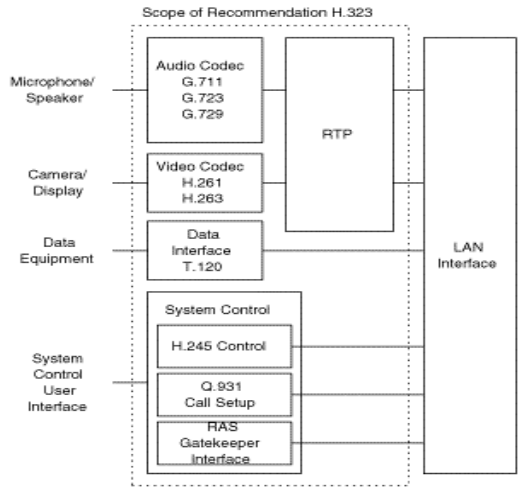
RTP (contd)

- RTP services: payload type identification, sequence numbering, timestamping, delivery monitoring, & optional mixing/translation. UDP for multiplexing and checksum services
- RTP does not provide: mechanisms to ensure quality-of-service, guarantee delivery or prevent out-of-order delivery or loss.
  - RTP sequence numbers allow receiver to reconstruct the sender's packet sequence, or to determine the proper location of a packet, eg, in video decoding, without necessarily decoding packets in sequence.

H.323

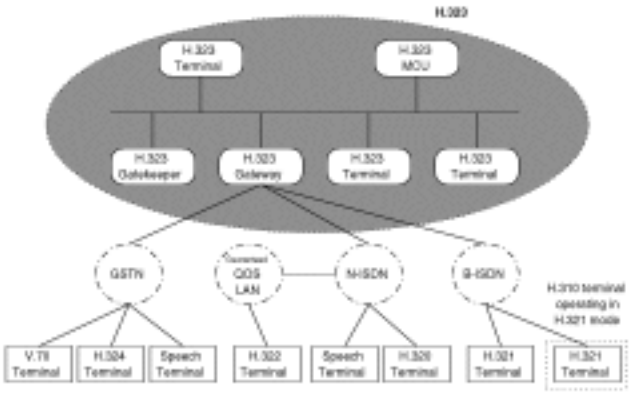
- H.323 is an ITU standard for multimedia communications over best-effort LANs.
- Part of larger set of standards (H.32X) for videoconferencing over data networks.
- H.323 includes both stand-alone devices and embedded personal computer technology as well as point-to-point and multipoint conferences.
- H.323 addresses call control, multimedia management, and bandwidth management as well as interfaces between LANs and other networks.

H.323 Architecture



H.323 (contd)

- Terminals, Gateways, Gatekeepers, and Multipoint Control Units (MCUs)



Renssel araman

H.323 (contd)

- Terminals: All terminals must support voice; video and data are optional.
- Gateway: an optional element which provides translation functions between H.323 conferencing endpoints (esp for ISDN, PSTN)
- Gatekeeper: most important component which provides call control services
- Multipoint Control Unit (MCU): supports conferences between three or more endpoints. Consists of a Multipoint Controller (MC) and Multipoint Processors (MP).

Rensselaer Polytechnic Institute Shivkumar Kalyanaraman

Integrated Services (int-serv)

- Supplement Internet Architecture with:
  - Services: guaranteed delay, controlled load
  - New signaling protocol: RSVP + admission control
  - *Shaping* at edge nodes combines with packet *classification* and *scheduling/buffer management* at routers to provide local delay and bandwidth guarantees.
  - Specs for parameters (flow-spec), classification (filter-spec)
- **Critique**: non-scalable, no control over routing vagaries, no feedback support

RSVP

- A *signaling* protocol: creates and maintains distributed reservation state
- *Multicast* trees setup by routing protocols, not RSVP (unlike ATM signaling)
- *Receiver-initiated*: scales for multicast
- *Soft-state:* time out unless refreshed: robust.
- Latest paths discovered through “PATH” messages and used by RESV mesgs.
- *Flowspec:* specifies resource to be reserved
- *Filterspec:* specifies how to classify packets
- Reservation styles: "wildcard", "fixed-filter", and "dynamic-filter".

Diff-serv motivations

- **#1.** ***Economics*** of ISPs (access and transit providers) dictates ***need for service differentiation***
  - IP provides just a best effort service
  - TOS is used in a non-standard way, and could be redefined to be more useful
  - Work done in pricing aspects of SLAs did not fit into IP because of a lack of header bits
  - ISPs, not IETF, should define services
    - Some services could be end-to-end, but here IETF would standardize only building blocks

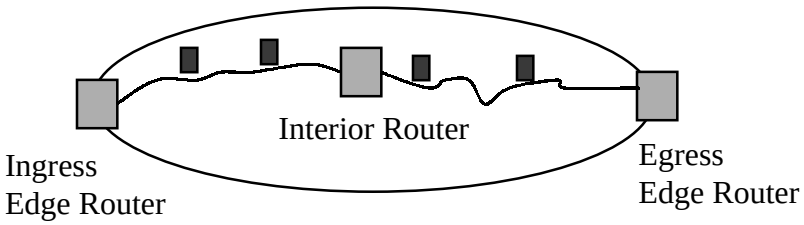
Diff-serv motivations (contd)

- **#2.** Diffserv is a considered to be ***crucial building block*** to provide performance assurances ***in IP-based VPNs.***
  - Other pieces: IPSEC (security & tunneling), L2TP (remote-access tunneling), and RSVP (QoS signaling)
- **#3.** ***Int-serv/RSVP does not scale***
  - Diff-serv uses a limited set of "behavior aggregates (BA)"
  - Diffserv creates a separation between edge and core routers.
    - Move per-flow (possibly non-scalable) data path functions (or MF-classification) to edges.
    - Edge handles policy, contracting and billing.
    - Interiors may participate in signaling

Diff-serv motivations (contd)

- ***Diff-serv must work with IPv4.***
  - Costs: incompatibility…
    - Redefining TOS octet.
    - Compatibility w/ RFC 791 (IP precedence)
    - New implementation of critical forwarding path as a "per-hop behavior"
  - Opportunities: leveraging Internet protocol base
    - Vendors: Opportunity for router upgrades
    - Small/medium-sized providers: economic necessity.
    - Large providers: view diff-serv as an intermediate solution to QoS while waiting for MPLS to integrate ATM, FR facilities and get traffic engineering features.

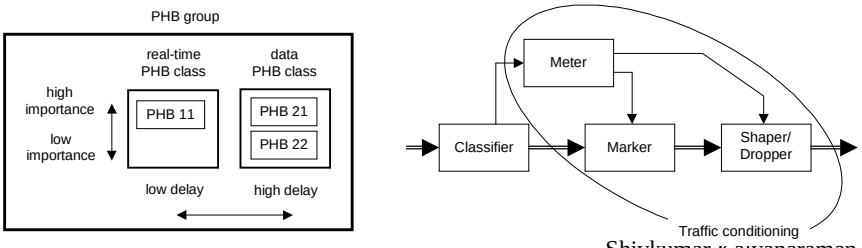
Differentiated Services Model



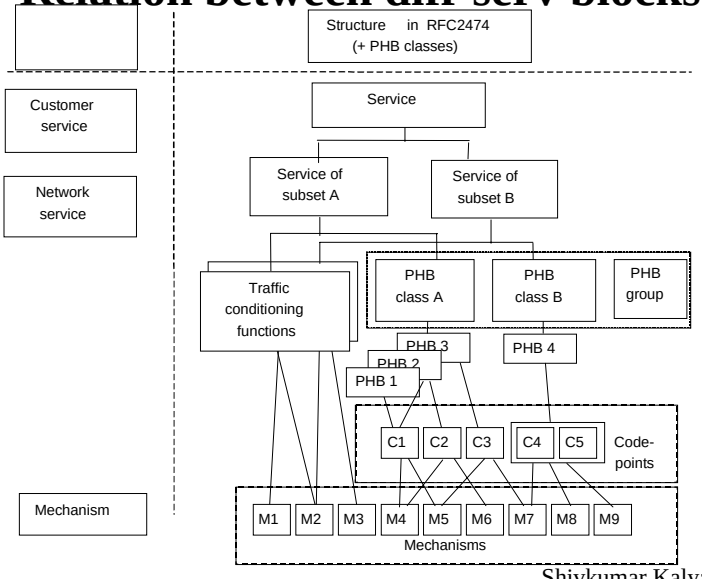
- ***Network edge routers:*** traffic conditioning (policing, marking, dropping), SLA negotiation
  - Set values in DS-byte based upon negotiated service and observed traffic. Per-flow state.
- ***Interior routers:*** traffic classification and forwarding
  - Use DS-byte as index into forwarding table

Diff-serv building blocks

- ***Per-hop Behavior: (PHB)*** generalization of mechanisms applied to a flow in the forwarding path
- ***PHB Group:*** Inter-related PHBs used together to implement a service.
- ***Codepoints:*** Bit combinations in the DS-byte
- ***Mechanisms:*** low level impln of building blocks
- ***Traffic conditioners:*** markers, meters, shapers etc



Relation between diff-serv blocks



IP Differentiated Services

- Only building blocks, no fully defined services
- Works with IPv4
- ***Services***: leased-line emulation("premium service"), frame-relay emulation ("assured service"), CoS (Class-of-Service)
- Only data-plane building blocks defined: traffic conditioners, Per-hop Behaviors (PHBs)
- **Critique:** control-plane components undefined (contenders: RSVP, COPS, SNMP, MPLS, L2TP)

Control plane: MPLS

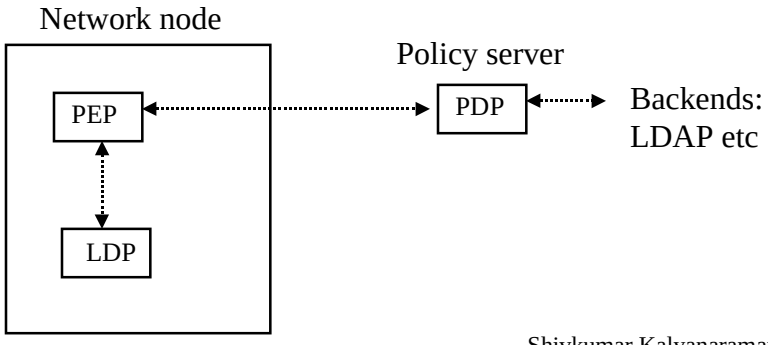
- Provides a framework for routing evolution
  - De-couples forwarding from routing control
  - Explicit routing
  - Constraint-based (QoS) routing, load-balancing
  - Traffic engineering: aggregating traffic flows into trunks, and mapping them onto pre-defined paths
- Provides a framework for integrating IP, ATM, and frame-relay cores
  - Allows re-engineering of the ATM control plane, and the IP forwarding plane

MPLS: building blocks

- ***Label***: short, fixed length field
- ***Forwarding table structure:***
  - Incoming label + subentry = outgoing label, outgoing interface, next-hop address (will include PHBs for diff-serv)
- ***Carrying label in header:***
  - Use VCI/VPI or DLCI in ATM or FR
  - New "shim" header for other link layers
- Forwarding algorithm: ***Label swapping.***
  - Use label as an index (exact match)
- ***Control component:***
  - Responsible for distributing routing & label-binding information: extensions to routing protocols, RSVP, LDP

COPS

- Common Open Policy Service
- Initially designed for adding ***policy control*** to RSVP
- Now being extended to support ***provisioning***
- Uses TCP; stateful exchange; common object model



Missing pieces in diff-serv

- ***Provisioning/policy/signaling:*** Assumed to be done using RSVP, COPS, SNMP, LDAP or over-engineering!
- ***Route pinning/multi-paths:*** extensions to OSPF, BGP, QoS routing
- ***End-to-end services:*** combination of above pieces: eg: frame-relay emulation, virtual leased line etc
- Tools to prevent traffic based denial of service attacks

Summary

- QoS big picture; ATM and IP building blocks/services
- Real-time transport/middleware: RTP, H.323
- Integrated services: RSVP, 2 services, scheduling, admission control etc
- Diff-serv: edge-routers, core routers; DS byte marking and PHBs
- Missing pieces: routing support (MPLS), pricing models, policy management (COPS)