Communication Networks II
Multimedia Communications / QoS
Specific Topics
(QoS, IntServ, DiffServ)

Prof. Dr.-Ing. **Ralf Steinmetz**

TU Darmstadt - Technische Universität Darmstadt,
Dept. of Electrical Engineering and Information Technology, Dept. of Computer Science
KOM - Multimedia Communications Lab
Merckstr. 25, D-64283 Darmstadt, Germany, Ralf.Steinmetz@KOM.tu-darmstadt.de
Tel.+49 6151 166151, Fax. +49 6151 166152

httc - Hessian Telemedia Technology Competence-Center e.V
Merckstr. 25, D-64283 Darmstadt, Ralf.Steinmetz@httc.de

Scope

<table>
<tr><td></td><td colspan="9">KN III (Mobile Networking), Distributed Multimedia Systems (MM I and MM II), Telecooperation II,III. ...; Embedded Systems</td></tr>
<tr><td rowspan="2">L5</td><td>Applications</td><td rowspan="2">Terminal access</td><td rowspan="2">File access</td><td rowspan="2">E-mail</td><td rowspan="2">Web</td><td colspan="2" rowspan="2">Peer-to-Peer</td><td rowspan="2">Inst.-Msg.</td><td>IP-Tel.</td></tr>
<tr><td>Application Layer
(Anwendung)</td><td>SIP & H.323</td></tr>
<tr><td>L4</td><td>Transport Layer
(Transport)</td><td colspan="3">Internet:
UDP, TCP, SCTP</td><td rowspan="3">Netw. Transitions</td><td rowspan="3">Security</td><td rowspan="3">Addressing</td><td colspan="2">Transport QoS - RTP</td></tr>
<tr><td>L3</td><td>Network Layer
(Vermittlung)</td><td colspan="3">Internet:
IP</td><td colspan="2">Network QoS</td></tr>
<tr><td>L2</td><td>Data Link Layer
(Sicherung)</td><td colspan="3">LAN, MAN
High-Speed LAN</td><td colspan="2"></td></tr>
<tr><td>L1</td><td>Physical Layer
(Bitübertragung)</td><td colspan="8">Queueing Theory & Network Calculus</td></tr>
<tr><td></td><td colspan="9">Introduction</td></tr>
</table>

Legend: **KN I** **KN II**

Overview

1. Motivation
   1.1 Quality-of-Service
   1.2 Repetition: Network Layer (Layer 3)

2. IntServ & Resource ReSerVation Protocol RSVP
   2.1 IntServ – Components
   2.2 IntServ – Service Classes
   2.3 The RSVP Protocol
   2.4 RSVP - Creating and maintaining reservation state
   2.5 RSVP – Merging of Reservations

3. DiffServ: Differentiated Services for the Internet
   3.1 DiffServ: Basic Ideas
   3.2 DiffServ: Proposed Services

4. Price-Controlled Best-Effort

5. Summary: IntServ, DiffServ, Price Controlled Best Effort, Best Effort

1. Motivation

Vision

INFORMATION SUPERHIGHWAY

Convergence of

- Internet
- telephony network
- radio and T.V. network
- ...

- **all wired and mobile**

One infrastructure for all (digital) services

⇒ **the** MULTI-SERVICE INTERNET

Multiservice Internet

Services on APPLICATION layer (applications):

- **today**
  - E-Mail
  - web
  - FTP
  - instant messaging
  - Peer-to-Peer file-sharing

- **next years (high-bandwidth, real-time applications)**
  - telemedia telephony (what about emergency calls?)
  - video (in acceptable quality)
  - network games

- **science-fiction (?)**
  - tele-medicine
  - highest quality immersive video everywhere
  - virtual worlds in real use
  - robot / car / ... control via Internet

Multiservice Internet (2)

Services on NETWORK layer:

- **best-effort service**
- **guaranteed service**
- **...**

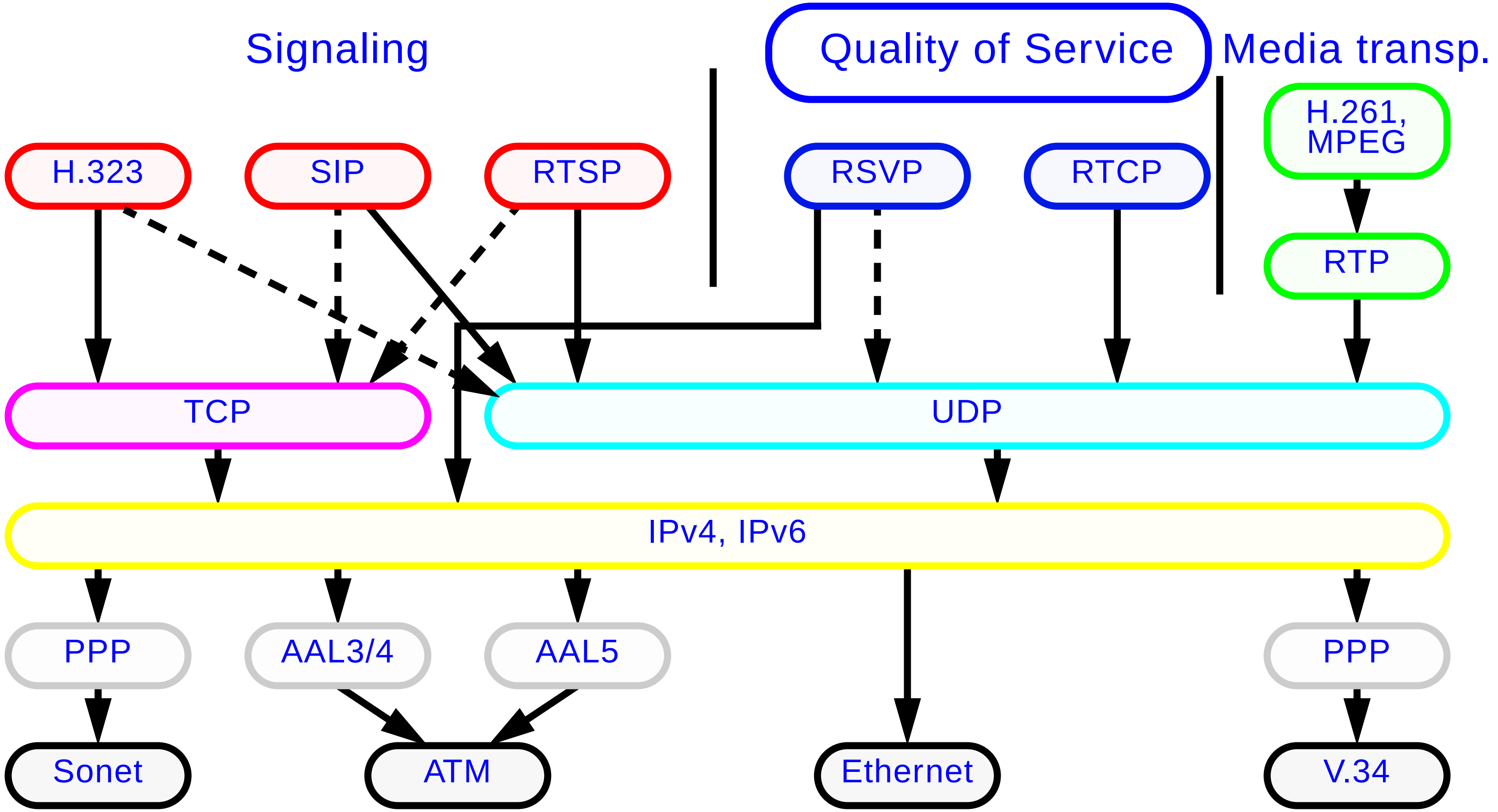
⇒ **see further discussion**

Currently only one service on network layer:

- **best-effort service**

⇒ **QUALITY OF SERVICE must be supported (somehow) at network layer**

Internet Real-Time and Multimedia Protocols



1.1 Quality-of-Service

Requirements of Different Applications

Continuous-media / discrete-media data presentation:

- real time requirements

Mode dependent:

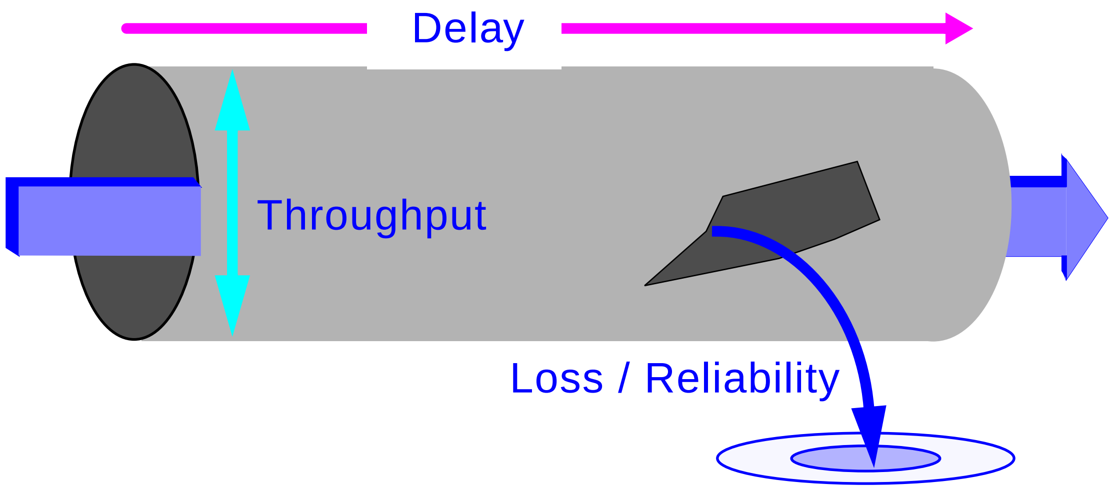
- off-line
- retrieval / distribution
- dialogue

Media and encoding dependent:

- discrete media / continuous media
- compressed / uncompressed / compression method

Affected parameters:

1. priority
   - delay / jitter
   - throughput
   - loss
2. priority
   - availability, security, ...



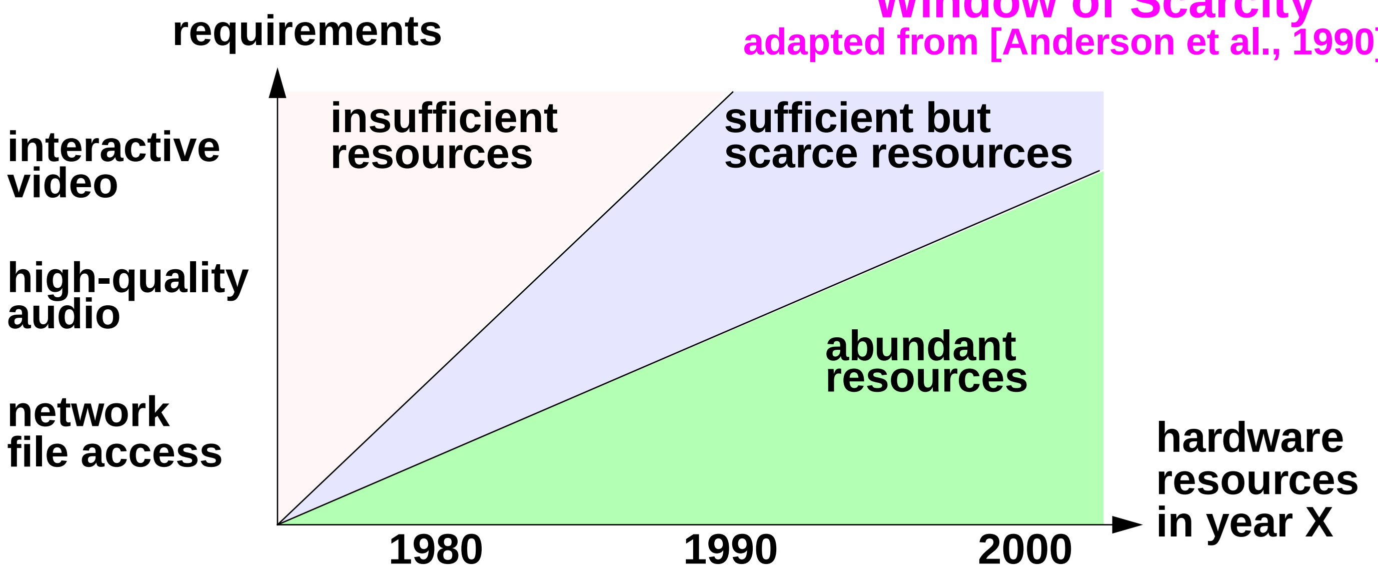
Why Resource Administration?

QoS depends on available resources

Resources and multimedia requirements:

- **always:**
  - **competition** for resources among tasks
  - desire to provide **best service at lowest possible costs**

"Window of Scarcity"
adapted from [Anderson et al., 1990]



⇒ **RESOURCE ADMINISTRATION to enforce QoS guarantees**

Quality-of-Service – Main Issues

QoS specification:

- **application’s requirements**
- **guarantees returned by the system**

QoS calculation:

- **functions to calculate QoS guarantees**

QoS enforcement:

- **reservation of resource capacities**
- **scheduling of resource access**

The 4 Approaches for Quality of Service

1. **IntServ (and RSVP)**
   - resource reservation (per flow) and admission control
   - queuing priorities based on flow

2. **DiffServ**
   - introduce a number of service classes
   - queuing priorities based on service class

3. **Price-Controlled Best-Effort (Congestion-Pricing)**
   - don’t change much
   - let users that cause congestion pay
     - ... and hope some of them back off

4. **Overprovisioning**
   - don’t change anything
   - just add enough resources (routers, bandwidth)
   - ... and pray

QoS Control
simplicity

more changes

1.2 Repetition: Network Layer (Layer 3)

Network layer protocol IPv4

- UNRELIABLE DATAGRAM SERVICE
  - NO FLOW control
  - NO ERROR control
- NO FIXED ROUTES
  - flexibility for path selection
  - reordering problems
  - ***NOT SUITABLE*** for time-critical continuous-media data
- **maximum datagram is 64 KByte**
  - segmentation for smaller subnets (e.g., Ethernet 1500 byte)
  - reassembly necessary (within endsystem)
- **checksum for IP header only (to avoid misdirection)**
- **Time-To-Live (TTL) = hop-counter to break loops**

$\Rightarrow$ **Modification of Internet protocols and mechanisms in order to provide QUALITY OF SERVICE**

Transport Layer (Layer 4)

TCP:

- congestion control included

UDP:

- no congestion control included

Today:

- most of the traffic is TCP (Web, Mail, Napster)

Probable Future:

- video and audio streams will increase UDP's share of total traffic

$\Rightarrow$ **(missing) Congestion control becomes more and more of a problem**

2. IntServ & Resource ReSerVation Protocol RSVP

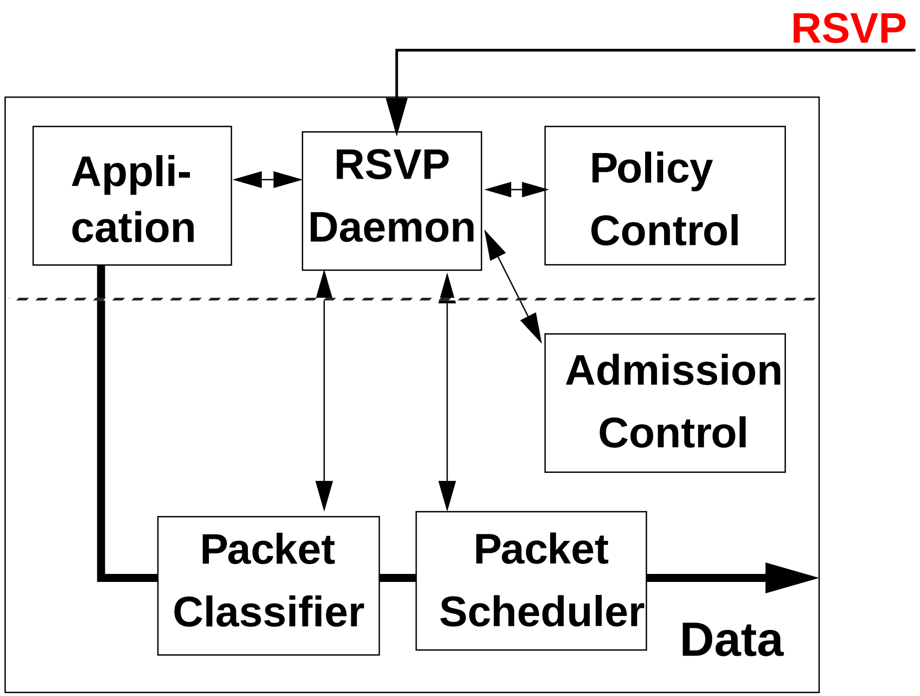
The 'Pure' Internet Model for QoS Provisioning

Use IP and IP Multicast for data transmission:

- **no new data forwarding protocol**

Additional mechanisms, e. g.:

- **reservation protocol**
  - Resource **R**e**S**er**V**ation **P**rotocol
    - RSVP
- **resource management modules**
  - e.g. admission control, packet classifier, scheduler



Integrated Services (Intserv)

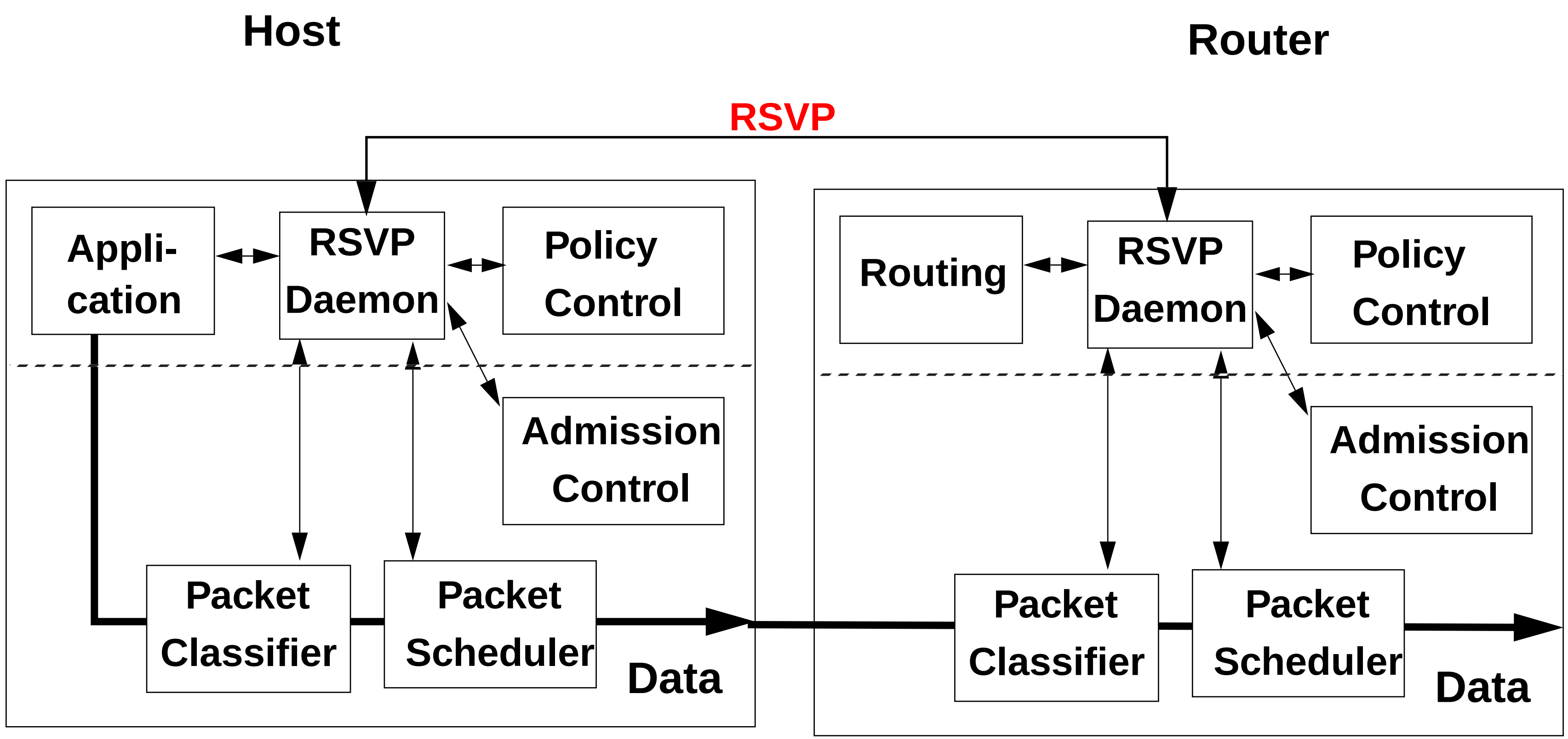
Framework developed with IETF Goal:

- **efficient Internet support for applications which require SERVICE GUARANTEES**
- **fullfil demands of**
  - **MULTIPOINT, REAL-TIME APPLICATIONS**
  - for
    - small and
    - large group communication
  - typical example:
    - large-scale video conferences

2.1 IntServ – Components

End-system and router components

- **existence and application of modules**
  - depends on specific service used



2.2 IntServ – Service Classes

3 service classes:

- **guaranteed service:**
  - throughput and delay guarantees
- **controlled-load service:**
  - limitation of load
  - similar to best-effort service in unloaded network
- **best effort:**
  - traditional IP service:
    - no limitations,
    - no guarantees,
    - no effort for QoS provisioning
  - default

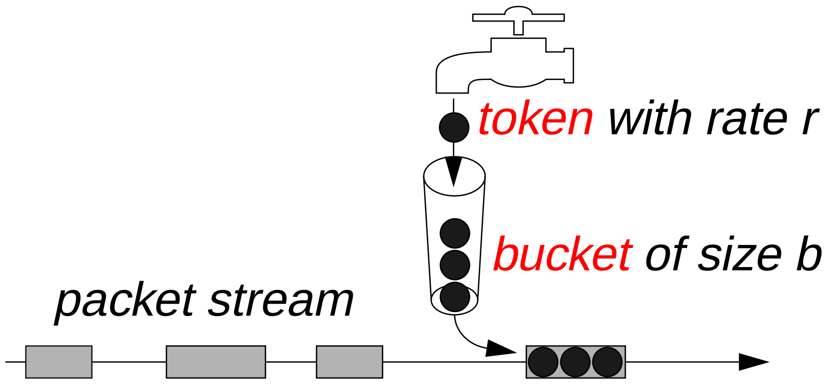
Additional classes (suggested, but postponed):

- **committed rate**
- **predictive delay**
- **controlled delay**
- **protected best-effort**

IntServ – Characterization of Traffic

Stream traffic characterized

by TOKEN BUCKET model



For

- **guaranteed and**
- **controlled-load service**

with

- **r = long-term rate (bytes/s)**
- **b = burst (bytes)**
- **M = Maximum packet size (bytes)**
- **m = minimum policed unit (bytes)**
  - minimum number of tokens required to send an IP packet
- **p = peak rate (bytes/s)**

Guaranteed Service

Strong guarantees:

- **guaranteed bounds for bandwidth and delay**
- **for applications with hard real-time requirements**

Required mechanisms:

- **admission control**
  - checks whether a new reservation request can be accepted
- **policing**
  - checks whether a flow conforms to its traffic description
- **reshaping**
  - adapts a flow to its traffic description
  - needed within the network to reduce bursts caused by jitter
- **per-flow scheduling**
  - determines the order by which packets are served
  - based on reservations and guarantees

Controlled-Load Service

Limitation of load

- **upper bound for the total traffic on the network**
- **no strong guarantees for bandwidth or delay**
  - weak assurance that
    - only a small percentage of the traffic is lost or delivered late
  - no quantification of QoS values
  - similar to best-effort service in unloaded networks
- **for applications that can adapt to moderate losses**

Required mechanisms:

- **admission control**
- **policing**

2.3 The RSVP Protocol

RSVP: Resource ReSerVation Protocol

- **RFC 2205 (September 1997)**
- **and more details at other RFCs**

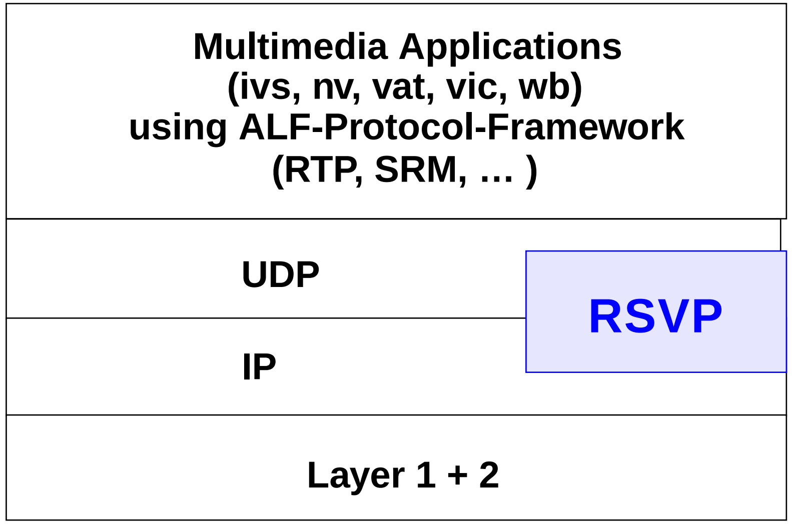
Contains

- **only protocol elements for control**
- **not for data transfer**

Companion protocol to IP

- **controls HOW IP SENDS A PACKET**
- **resource reservation support**

RSVP in the Protocol Stack



Typical environment with

- **resource reservation protocol (RSVP)**
- **simple transport protocol (UDP)**
- **Application Level Framing (ALF):**
  - integration of protocol framework into application
    - (RTP: real-time transport protocol,
    - SRM: scalable reliable multicast)

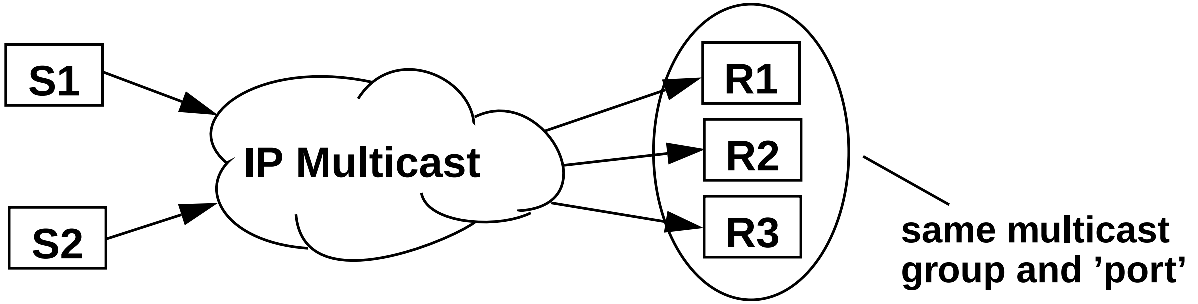
RSVP – Basics

Main abstractions:

- **IP multicast routing tree from source(s) to multiple targets**
- **receiver-initiated reservation**
- **filtering provides for**
  - heterogeneous receivers
  - different reservation styles
- **concentrates on resource reservation only**
- **‘soft-state’, refreshed periodically**

RSVP Flow and Session

Simplex transmission model



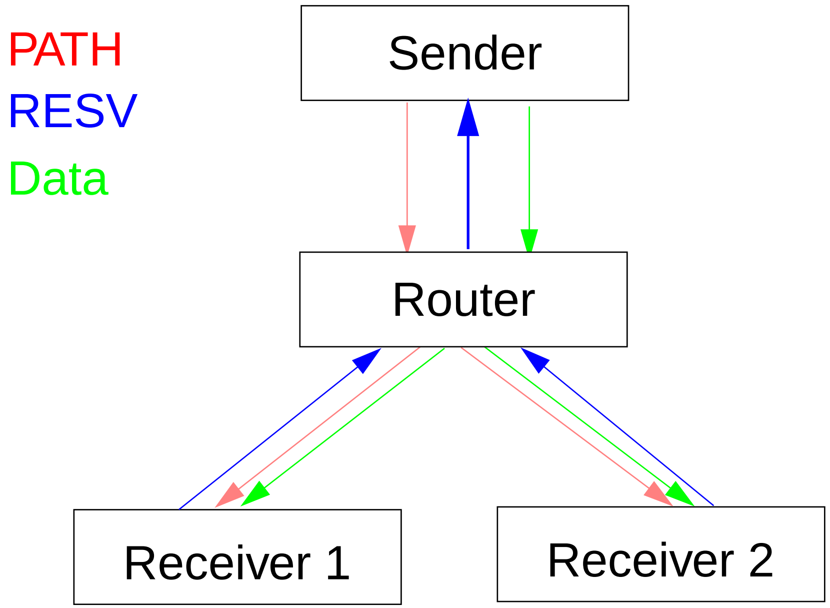
Sessions:

- **destination address**
  - unicast or multicast
- **reservation ID (32 bit number)**
  - generalized receiver ’port’; supplied by application (’cookie’ for RSVP)
- **protocol number**
- **1+ flows**

Data flows distinguished by

- **IPv4: source IP address, source port**
- **IPv6: source IP address, flow label**

IP Multicast with Resource Reservation (RSVP)



Periodic transmission of

- **PATH message indicating session parameters**
  - sent from sender to complete group
- **answer: reservation message (RESV) from receivers to sender**
  - use route defined through PATH messages
  - receiver-initiated reservation
- **routers reserve resources based on RESV messages**
- **soft state update**

2.4 RSVP - Creating and maintaining reservation state

Source/sender:

- **multicasts data flows**
- **sends PATH message (periodically) including TSpec describing flows**

Receiver:

1. **joins multicast group**
2. **receives PATH messages**
3. **determines own QoS requirements and uses received TSpec**
4. **sends RESV message including filter and flow spec for each sender's flow**

- **periodic refresh of 'soft-state' via transmission of**
    - **PATH** messages
    - **RESV** message
- **reservations are only valid for a certain time and TIMEOUT**
- **source not restricted from transmitting data at any time**
- **packets may go across unreserved routes**
- **forwarding protocol must be aware of**
    - relation between packets and reserved resources

Receiver-Initiated Reservations – Reasons

Dynamic membership:

- **endsystems join and leave transmissions frequently**
- **if sender initiated then sender must handle these messages**
  - potential overload

Large group size:

- **receiver-initiated scheme reduces sender load**
- **merging of reservations**

Heterogeneous receivers:

- **world is heterogeneous**
  - networks,
  - endsystems
- **receiver knows its requirements best**

Receiver-Initiated Reservations – Drawbacks

Good reasons, but not always true / applicable

Moreover, receiver-orientation leads to other problems, e.g.:

- **RSVP background is large-scale conference**
  - this is just one application type
  - suitability for many small-scale applications?
    - e.g., video-conferences, VoD, Internet-Phone ?
- **heterogeneous flows must be supported not only heterogeneous reservations**
  - example:
    - merging of 1 MBit stream with a 100 MBit stream.
    - random drop of 99% of the packets?!
    - filtering on data path necessary,
      but, too expensive
- **routing based on QoS characteristics very difficult**
  - path set before reservation requirements are known

RSVP Messages

RSVP messages are

- **sent as datagrams directly over IP**
- **periodically resent:**
  - to refresh reservation state
  - to substitute lost messages

Message types

- **PATH**
- **RESV**
- **error messages (PathErr, ResvErr)**
- **teardown messages (PathTear, ResvTear)**

RSVP – Flow Descriptor

Flow Descriptor = (Q, F)

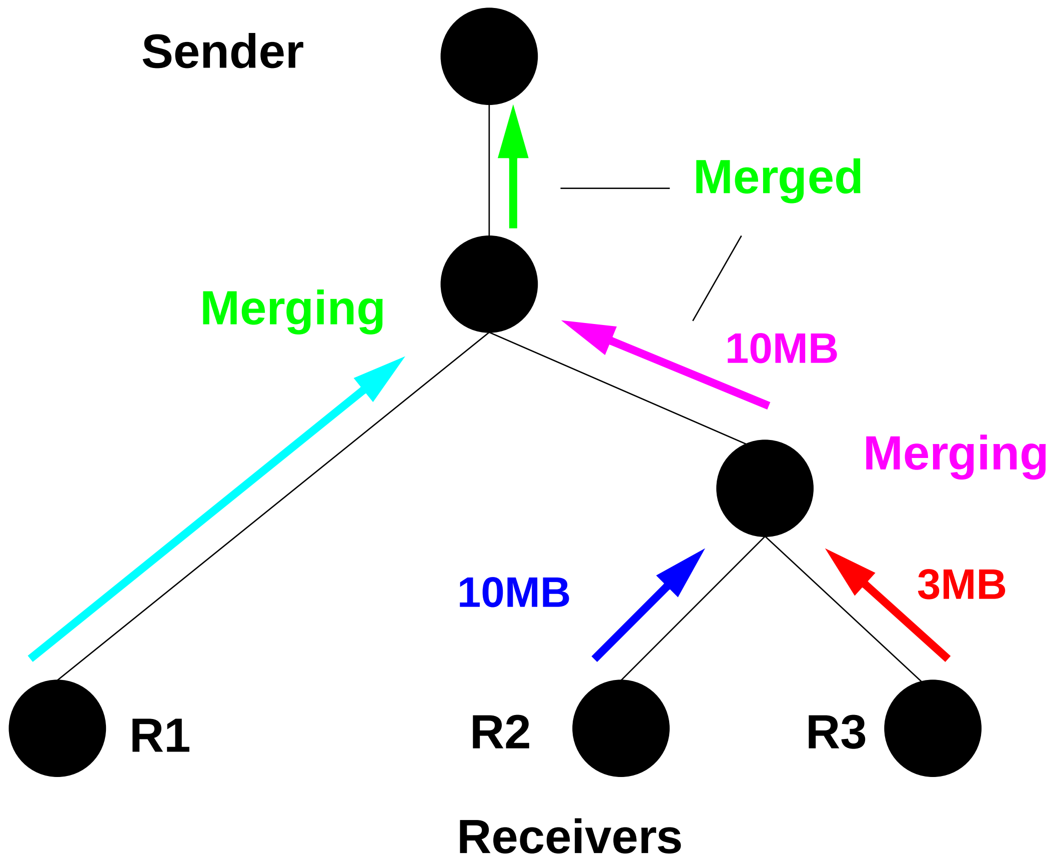
Q = *FLOWSPEC:*

- **defines desired QoS**
- **TSPEC:**
  - source behavior, leaky bucket
- **RSPEC*:***
  - reservation,
  - e.g. delay or priority

F= *FILTERSPEC:*

- **controls classifier**
- **to select the subset of data packets to receive this QoS**

2.5 RSVP – Merging of Reservations



Reservation Styles

Sender Selection	Reservation	
	Distinct	**Shared**
Explicit	Fixed-Filter (FF) style	Shared-Explicit (SE) style
Wildcard	undefined	Wildcard-Filter (WF) style

Shared vs. distinct reservations:

- **some applications can share a reservation among multiple sources**
  - e.g., usually only one person is speaking in a conference
- **other applications need one distinct reservation per source**
  - e.g., for video from all persons in a conference

Explicit vs. wildcard sender selection:

- **some applications want to make reservations for explicit senders**
  - e.g., teleteaching
- **some applications want to make reservations for any sender**
  - e.g., conference

RSVP – Filters

Originally specified filters:

wildcard	no-filter mode, sender's flow is not filtered (all senders share the reserved resources)
fixed	sender's flow filtered according to a fixed filter during reservation (only single sender can use reserved resources)
shared explicit	set of specified senders share the reserved resources

Merging – Fixed-Filter Style

s*: senders

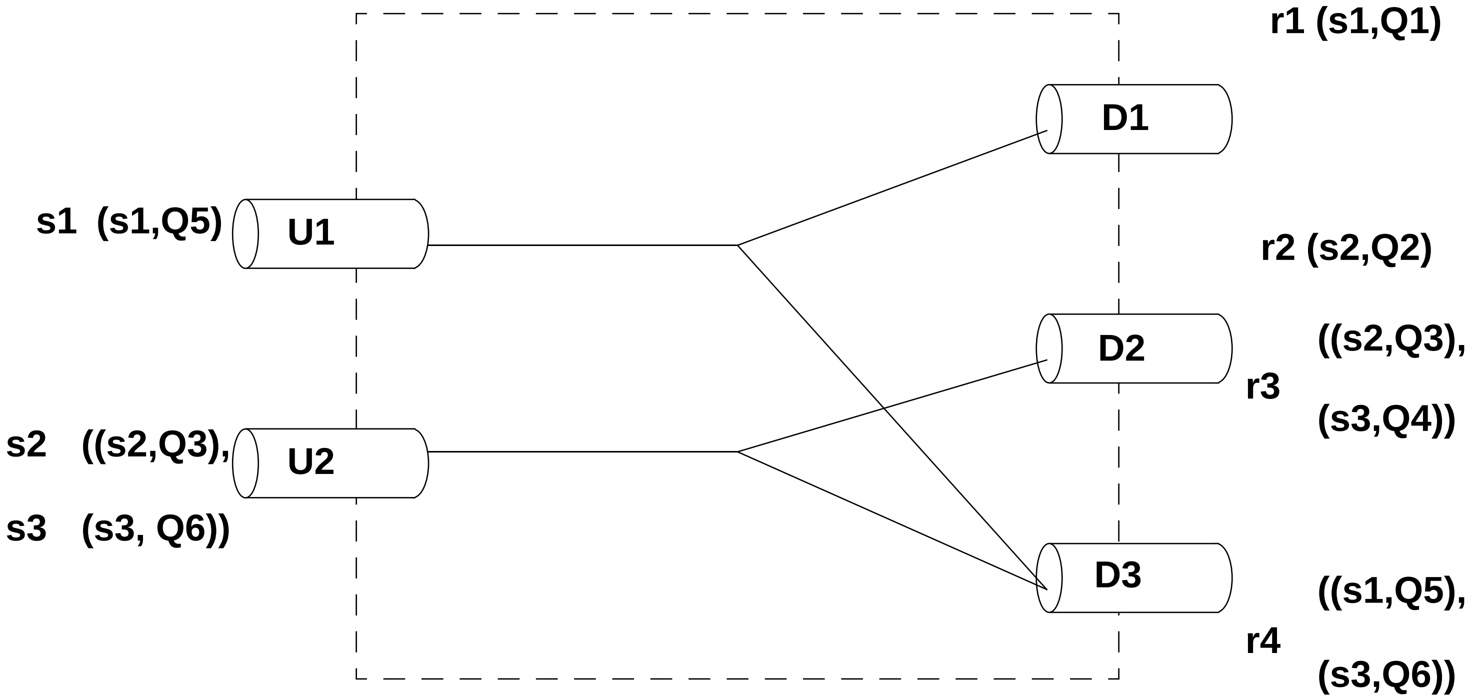
r*: receivers

Q*: FlowSpec

U*: upstream interfaces

D*: downstream interfaces

assume: Q1 < Q2 < Q3 < Q4 < Q5 < Q6



- **each interface reserves**
  - maximum of received reservations for each source
- **separate reservation sent to each requested source**

Merging – Shared-Explicit-Filter Style

s*: senders

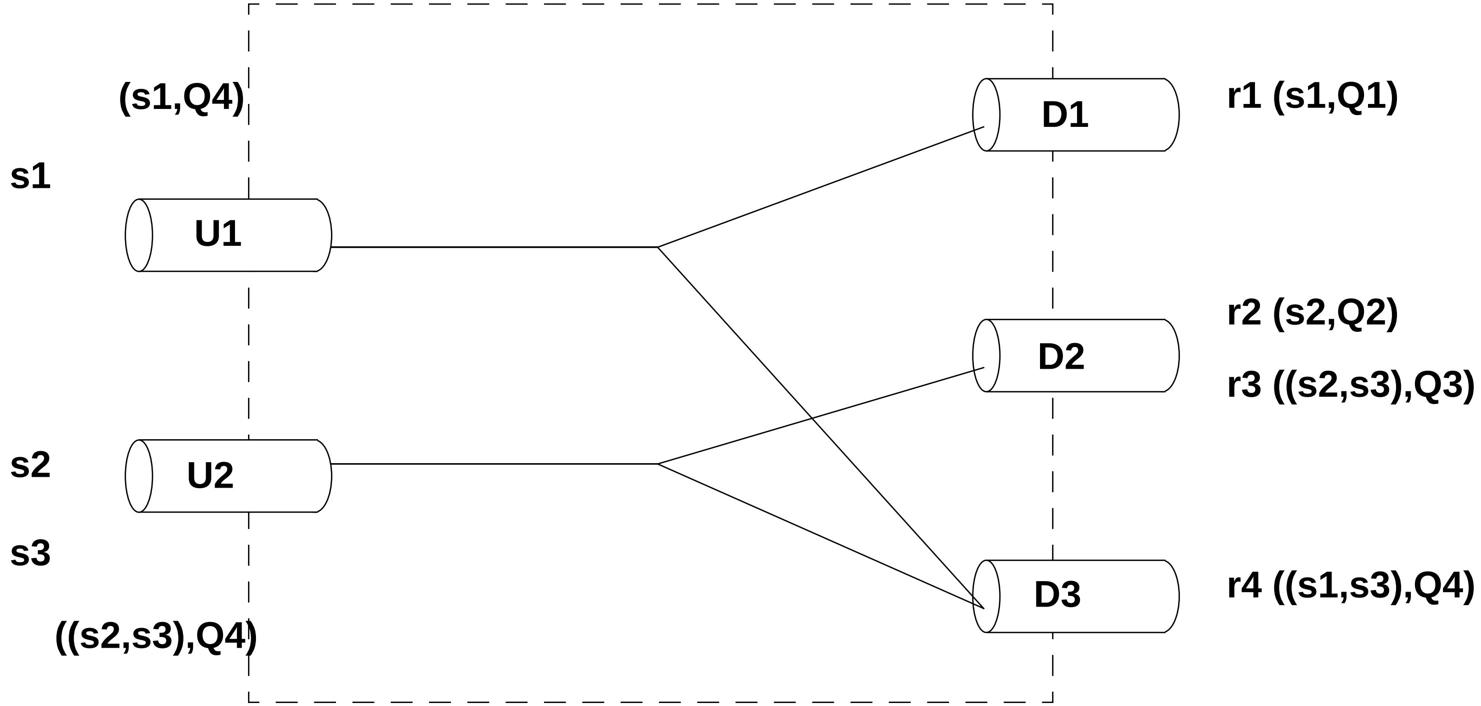
r*: receivers

Q*: FlowSpec

assume: Q1 < Q2 < Q3 < Q4 < Q5 < Q6

U*: upstream interfaces

D*: downstream interfaces



- **FilterSpec of merged reservations is union of FilterSpecs**
- **FlowSpec of merged reservations is maximum FlowSpec**

Merging – Wildcard-Filter Style

s*: senders

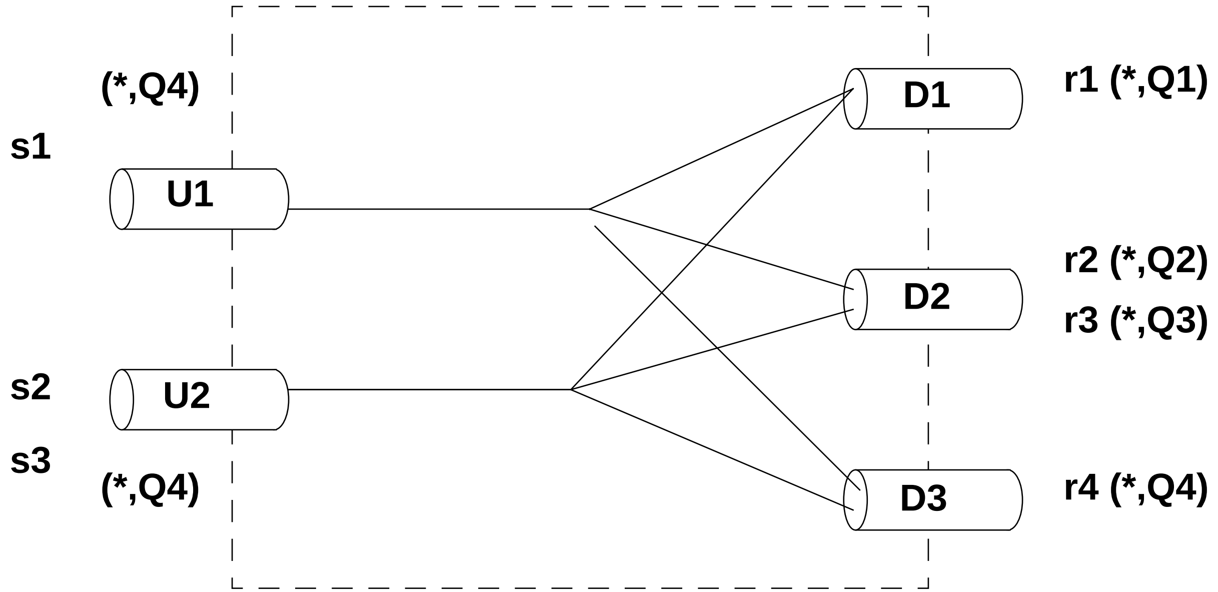
r*: receivers

Q*: FlowSpec

U*: upstream interfaces

D*: downstream interfaces

assume: Q1 < Q2 < Q3 < Q4 < Q5 < Q6



- **each interface reserves maximum of received reservations**
- **maximum of all reservations is sent to all sources**

RSVP, Routing and Soft-States

Data forwarding tree is set up by routing protocol

⇒ **RSVP and routing are decoupled:**

- simple handling of link failures
- route flapping possible
  - ".. The key to whether use of BGP will scale to a very large Internet lies in the stability of inter Autonomous System routing. If routes between Autonomous Systems vary frequently a phenomenon termed **FLAPPING** then the BGP routers will spend a great deal of their time updating their routing tables and propagating the routing changes.."
- no hard QoS guarantees

RSVP SOFT STATE MANAGEMENT

- **reservation is set for certain time only:**
  - ***REFRESH*** by end systems necessary
  - state ***TIMES OUT*** if no refresh received and reservations are removed
- **periodic transmission of:**
  - PATH from sender
  - RESV from receiver
  - merging at routers possible (e.g., RESVs from multiple receivers)

3. DiffServv:

Differentiated Services for the Internet

Background: scaling problems of IntServ

- **administration of each INDIVIDUAL flow**
- **huge overhead in large-scale networks**
- **recommendation:**
  - use IntServ for
    - small closed networks
    - limited amount of (perhaps concatenated) flows

DiffServ:

- **avoids drawbacks of best-effort and IntServ**
  - no strong guarantees
    but better service than best-effort (= no QoS management)
  - no management of individual flows,
    i.e. less overhead
- **minimalistic approach**
  - with regard to standardization
- **compatibility to IPv4**

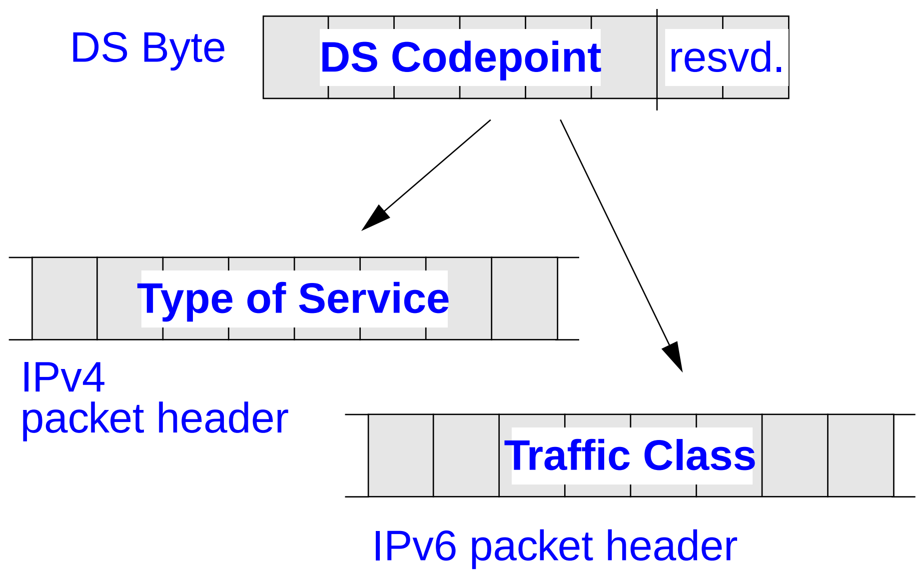
3.1 DiffServ: Basic Ideas

Aggregation of flows

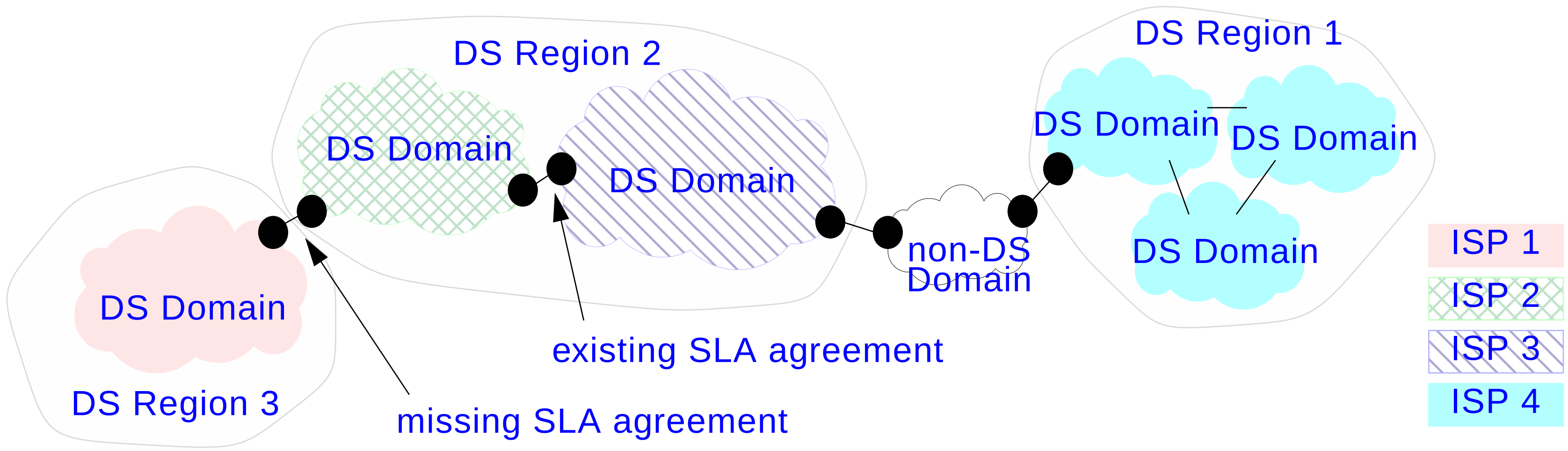
- **reservations for a group of related flows**
  - e.g. all flows in the same (priority) class
- **reservation of a more static nature**
  - for a longer period than a flow's lifetime

Tagging of IP packets

- **DS byte in packet header**
  - type of Service byte in IPv4
  - traffic Class byte in IPv6
- **determines treatment of**
  - packets within routers
  - e.g. packet priority
- **allows user and/or service provider**
  - to tag packets that shall be treated with preference



DiffServ Architecture Example



ISP: **Internet Service Provider**

SLA: **Service Level Agreement**

DS: **DiffServ**

3.2 DiffServ: Proposed Services

Expedited / Assured Forwarding

PHB = Per-Hop-Behavior (behavior inside one router)

- **absolute bandwidth allocated to aggregate flows**
- **DS byte specifies packet (priority) class**
  - Expedited Forwarding (EF),
  - Assured Forwarding (AF),
  - Best-Effort

PDB = Per-Domain-Behavior (behavior inside on provider’s domain)

- **virtual wire (Expedited Forwarding Per-Domain-Behavior EF PDB),**
- **assured PDB**
- **best-effort PDB**
- **bulk-handling PDB**

Service (offer by provider to customer)

- **Premium Service**
- **Assured Service**
- **Best-Effort Service**

DiffServ: Premium and Assured Service

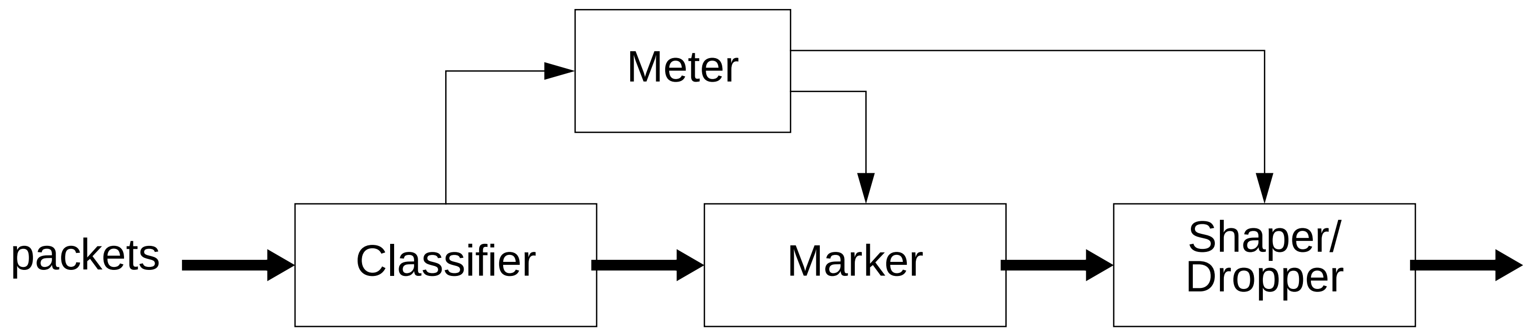
Premium Service:

- **contract between user and network**
  - source and target addresses of an aggregate flow
  - bandwidth available for the flow
- **similar to virtual leased line**

Assured Service:

- **no guaranteed bandwidth**
- **assurance that a high percentage of the flow will obtain a good service**

DiffServ: Premium and Assured Service (2)



Routers:

- **classification of packets**
  - management of P- and A-Bits
- **policing and shaping of flows**
  - based on token buckets
- **scheduling of packets**
  - high-priority queue for Premium Service
  - low-priority queue for Assured Service and best-effort packets
  - dropping of best-effort packets with a higher probability than assured packets

DiffServ: Other Proposed Services

Other proposed service models / pricing schemes:

- **Olympic Services:**
  - gold,
  - silver,
  - bronze
- **Paris-Metro Pricing:**
  - 1st and
  - 2nd class
- **Cumulus Pricing Scheme, ...**

4. Price-Controlled Best-Effort

Drawback of IntServ and DiffServ:

- modifications of Internet routers necessary
- IntServ and DiffServ Routers are more expensive

Alternative Idea

- stick to current best-effort services

Advantage

- best-effort (can be) good enough if there is no congestion
- best-effort routers are already deployed

Congestion in the future

- more and more UDP traffic
- UDP has no congestion avoidance mechanism like TCP

$\Rightarrow$ **new congestion avoidance mechanism necessary**

Basic Idea

Congestion avoidance

- **let users pay if they**
  - cause congestion
  - use already congested links,
- **so**
  - they have to decide whether
    - it is worth to stay and pay or
    - to reduce the amount of traffic

Fairness

- **bandwidth is allocated proportional to the willingness-to-pay**
  - proportional fairness
- **is this “fair”?**

Realization

Explicit Congestion Notification (ECN)

- **see RFC 2481 (for TCP/IP)**
- **and later**
  - RFC 3168 with different approach, for IP

Explicit Congestion Notification ECN

- **when a router experiences congestion**
  - it marks a number of random packets
    - marking a packet is done
      by setting the ECN bit in the TOS byte of the IP Header (IPv4)
- **the receiver informs the sender**
  - via TCP ACKs of incoming ECN signals
- **the sender reacts by**
  - decreasing its traffic

Use with/for Pricing

Using Explicit Congestion Notification ECN for pricing:

- **users pay a small amount of money per ECN mark they receive**
- **this gives them an incentive to decrease their traffic but does not force it**

Highly dynamic prices

- **users cannot predict prices**
- **prices can vary rapidly within seconds**

  ⇒ **Risk Broker**

Security

- **what can stop a provider marking too many packets**

  ⇒ **High Competition**

related to (price-controlled) Best-Effort

- **difficult to set the prices to the right magnitude**
- **will probably not be good enough for some applications**

Price Controlled Best Effort is not an Internet draft yet.

- **Research**
  - e.g. in the EU funded **M3I** Project and dfn project LetsQoS

5. Summary: IntServ, DiffServ, Price Controlled Best Effort, Best Effort

most interesting multimedia applications are networked

- **traffic requirements are**
  - very different from traditional data traffic

QoS control is an essential element of multimedia networking:

- **description**
- **negotiation**
- **provision**

4 approaches

- **IntServ (RSVP)**
- **DiffServ**
- **Price-Controlled Best-Effort (using ECN marks)**
- **Overprovisioning**

combinations of those possible / make sense

IntServ vs. DiffServ

Internet Integrated Services

- **ratio**
  - limited resources
  - hard quality requirements
- **approach**
  - resource reservation
  - per connection / flow
- **method**
  - distributed signaling protocol
  - router identify flows
  - scheduling
- **services**
  - best effort service
  - controlled load service
  - guaranteed service
- **scalability for large-scale net. ?**
  - packet classification in core router

Internet Differentiated Services

- **ratio**
  - abundance of resources
  - adaptive data streams
- **approach:**
  - aggregation of flows
  - reservations for aggregates
- **method**
  - static control
  - packets marked with priorities
- **services (suggestions)**
  - premium, expedited forwarding & assured (max. packets high prio)
  - “Olympic”: (Gold 60%, Silver 30%, Bronze 10% capacity)
- **no hard deterministic guarantees**
  - packet classific. at net. borders

Coexistence IntServ and DiffServ

IntServ: for small, closed networks

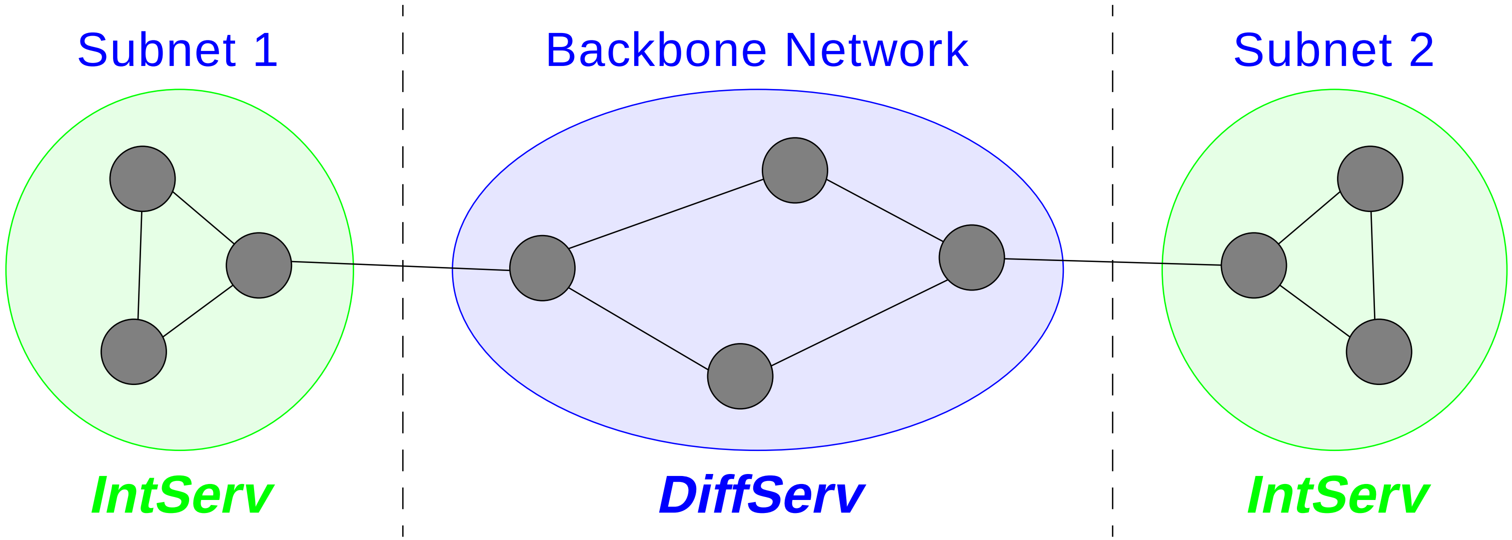
- **e.g. VPN (Virtual Private Network), Corporate Network**

DiffServ: for large, open networks

- **e.g. backbones**

$\Rightarrow$ **IntServ and DiffServ are not necessarily competing**

Integration is possible:



IntServ, DiffServ and Price Controlled Best Effort

Combination

- **Price Controlled Best Effort with ECN**
  - in the backbone
- **IntServ or DiffServ**
  - in the access network