# Differentiated Services:

A Tutorial Overview with a Voice over IP Slant

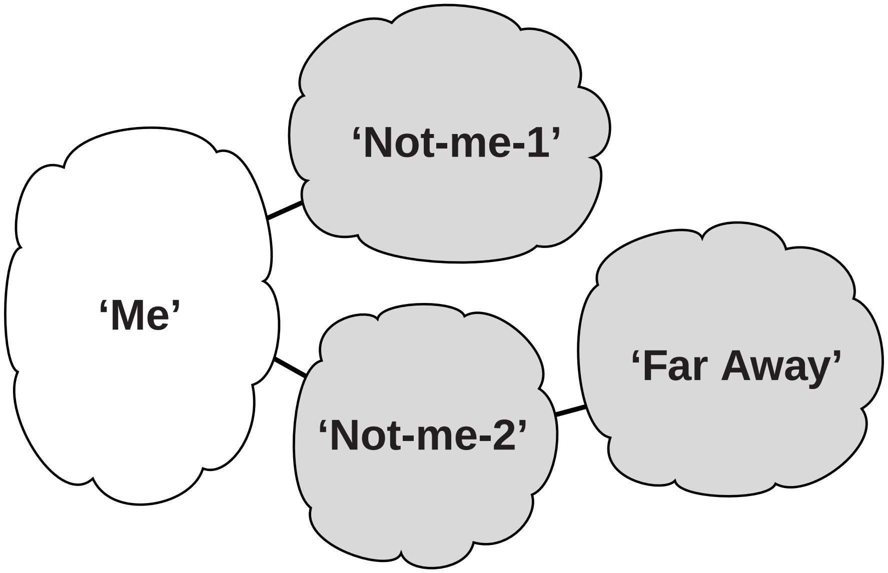
Kathleen Nichols
kmn@cisco.com

ETSI Workhop on Voice over IP
June 9, 1999

# Differentiated Services

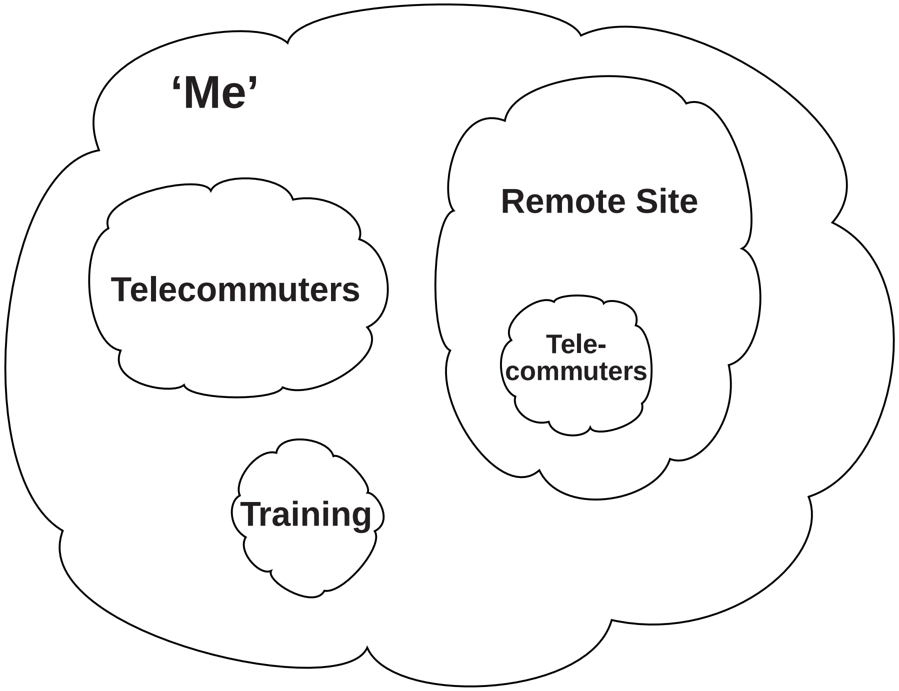
- The differentiated services architectural model is an approach to delivering QoS in a scalable, incrementally deployable way that:
  - keeps control of QoS “local”
  - pushes work to the edges and boundaries
  - requires minimal standardization, encourages maximal innovation
- The IETF Differentiated Services WG is working on the “minimal standardization” part of this
- Diffserv’s model is based on an Internet made up of independently administered domains, each of which is connected to at least one other

# An Architectural Framework based on Clouds and Boundaries



Follows the structure of today’s Internet: Clouds are regions of relative homogeneity in terms of administrative control, technology, bandwidth, etc.

# Clouds within Clouds



# QoS and Clouds

- Within a cloud, QoS is allocated according to some locally determined set of rules
- Almost all the work is confined to the boundaries of clouds and covered by a set of rules
- Rules might not be symmetric across a boundary
- QoS information exchanged between clouds is confined to boundaries and covered by bilateral agreements where clouds have different owners

# Advantages of Model Based on Clouds

- Clouds can map to the independently administered regions of the Internet
- Architecturally agnostic: within a cloud any technology might be used to deliver QoS
- Signaling agnostic and signaling can develop and evolve
- Possibility of multiple paths increases reliability
- QoS can be deployed in only one cloud, doesn’t need to be signaled per connection, and the state in most nodes can be reduced considerably as compared to connection-oriented approaches which tie up resources, require state for every connection and are not incrementally deployable or scalable

# Scalability through Aggregation

- Fundamental to the diffserv approach is:
  - › there are a relatively small number of ways to handle packets in the forwarding path
  - › the number of traffic conversations requiring QoS may be quite large and subject to a wide range of rules which devolve from policy
- Packets are grouped by the forwarding behavior they are to receive within a cloud
- Nodes in the center of a network only have to deal with the small number of traffic aggregates rather than keeping track of every separate traffic conversation that passes through

# Aggregation and Conversations

- The per-conversation state is kept at the edges
- Flows or conversations are classified into aggregates and are “conditioned” to meet the rules of that aggregate
- Packets are not marked for  the “services” individual conversations may be receiving. Many services may use the same marking. Any viable service must make sense under aggregation
- Don’t distinguish between flows, so the treatment the behavior aggregate receives should not result in different performance for different traffic compositions of the behavior aggregate

# “Minimal Standardization”: IETF RFC 2474

- A bit-field in the packet header determines the packet’s forwarding treatment. DS field is the TOS octet in IPv4 and the Traffic Class octet in IPv6; within that uses bits 0-5 as a “codepoint” field (DSCP)
- Codepoints should be looked at as an index into a table of packet forwarding treatments at each router.
- This table maps a DSCP to a particular forwarding treatment or “per-hop behavior” (PHB) that is applied to that behavior aggregate. PHBs are constructed by vendors from, for example, particular queue schedulers
- Behavior for only a few codepoints to be globally assigned and diffserv-capable equipment must make codepoint to behavior mapping flexible and accessible
- Class Selector Compliant PHBs get DSCPs 000000-111000

# The DSCP and PHBs in Use

- Diffserv-capable equipment will include packet classifiers (both DS field and multiple fields of the IP packet header) and configurable traffic conditioners such as shapers, policers, (re)markers. These can be used to control both selection and entry into a particular queue
- A packet's DS field may be marked with a codepoint "anywhere" in the network (but marking expected to occur at edges and boundaries)
- All packets with the same codepoint get the same behavior thus providing aggregation and scalability (a "behavior aggregate")
- Marking can be based on microflow identification, the packet ingress link, the measured temporal characteristics of a microflow or aggregate, etc.

# The EF Per-Hop Behavior (RFC 2598)

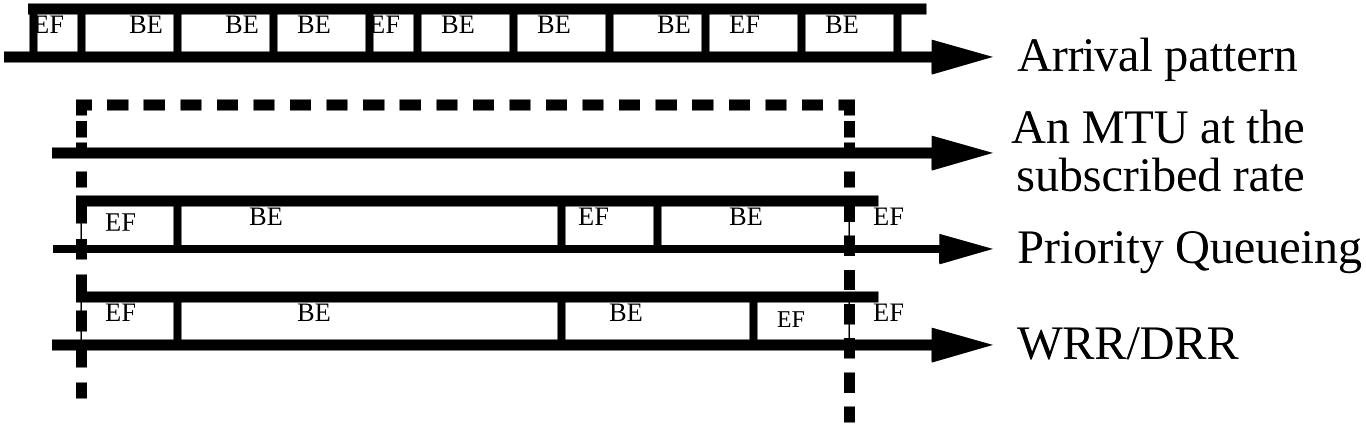
This is a forwarding behavior of general use, but it is also the one most useful for doing VoIP

In simple terms, it is a rough equivalent of PQ, can be implemented by PQ with some safety mechanism

More precisely (from RFC 2598)

> “the departure rate of the aggregate’s packets from any diffserv node must equal or exceed a configurable rate.
>
> The EF traffic SHOULD receive this rate independent of the intensity of any other traffic attempting to transit the node.
>
> It SHOULD average at least the configured rate  when measured over any time interval equal to or longer than the time it takes to send an output link MTU sized packet at the configured rate. “

# Example Implementations of EF PHB

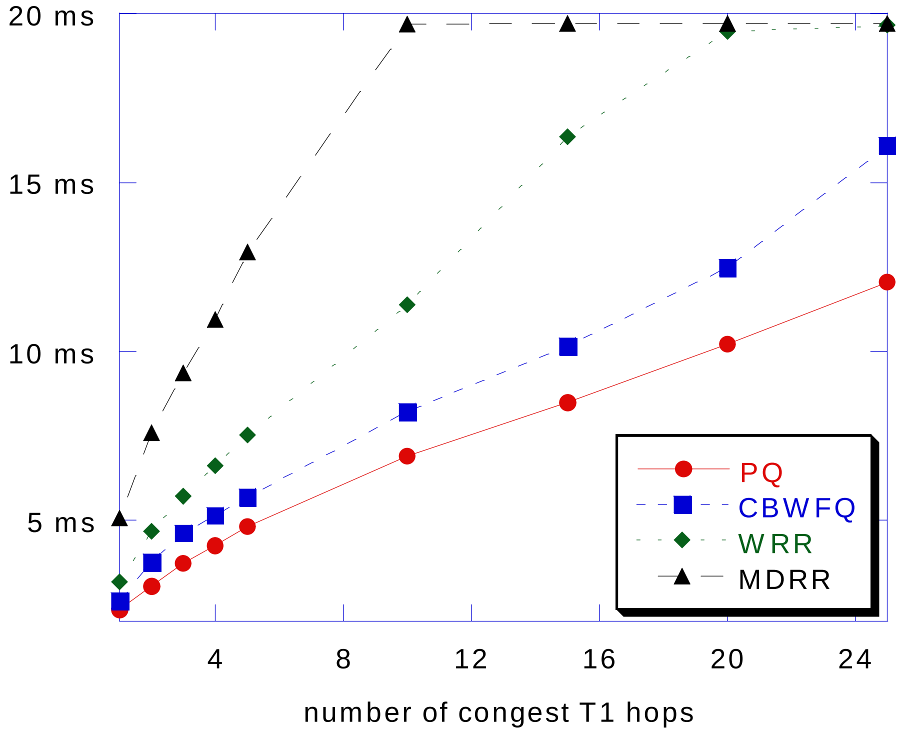


Arrival rate is twice the output link bandwidth

EF queue configured to get 25% of the output link bandwidth. BE is configured to get 75% of the output link bandwidth.

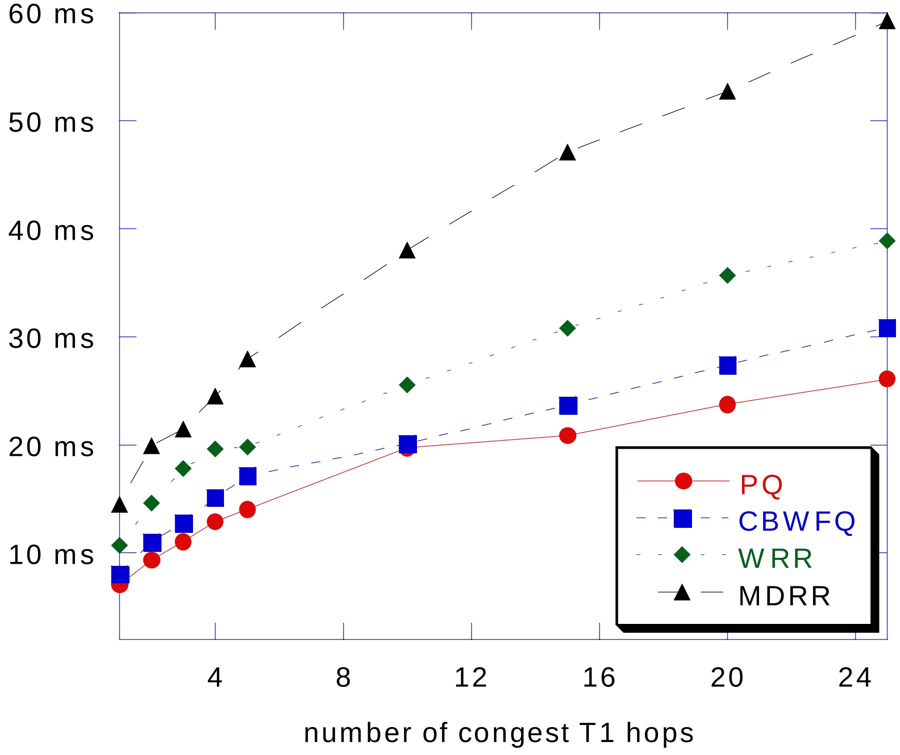
Here packets come in only two sizes, the MTU and one-half the MTU.

# Performance of VoIP using Diffserv's EF PHB



Median Jitter (|($a_j$-$a_i$)-($d_j$-$d_i$)|) for voice type packet streams. 10% of traffic is EF-marked, 60% gets other "special" treatment. Of the EF packets, half are long (1500 bytes), half short (100 bytes). (Results due to Y. Kim of Cisco)

# Performance ofVoIP using EF PHB (cont)

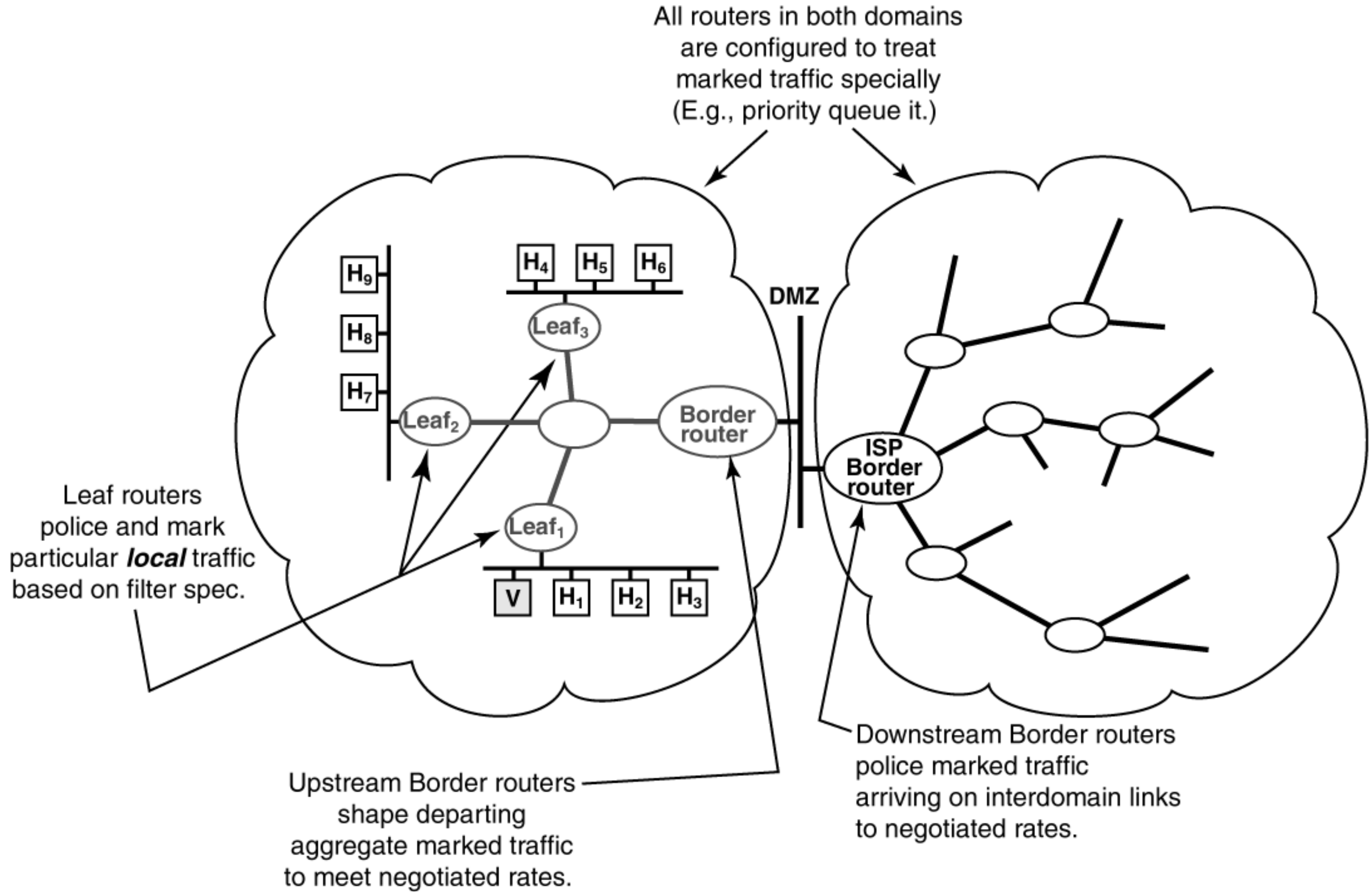


95th percentile of jitter

# Services in the Diffserv Framework

- That covers the forwarding path, but what about building services?
- Services are built by adding rules to govern behavior aggregates:
  - › initial packet marking
  - › how particular aggregates are treated at boundaries
  - › temporal behavior of aggregates at boundaries
- Different user-visible services can share the same aggregate
- Services must be sensible and quantifiable under aggregation

# Example: Putting Together an EF-based Service

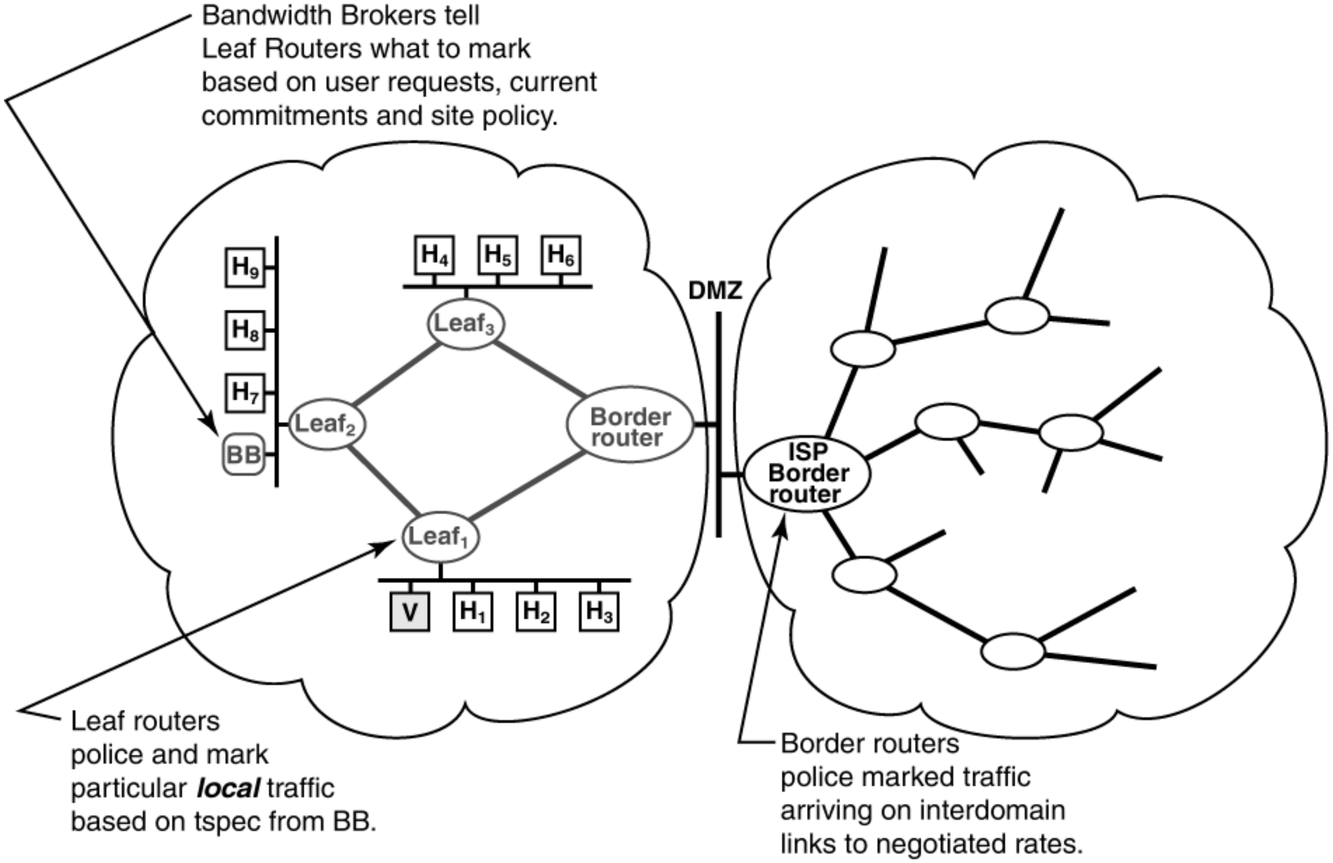


**Natural Question: How do the border and edge routers know what to mark?**

# Controlling the Boundaries

- A repository of of policy is needed to keep track of priorities and limits on QoS allocations for individual users, projects, and/or departments.
- An entity needs to receive requests for QoS, consult and update the database, and send configuration information to the routers, where indicated.
- Call this entity a “bandwidth broker” (BB) (Van Jacobson).
- BB is part of the network infrastructure and must authenticate requests from users. Some information can also be configured.
- Intradomain policy decisions and implementations remain up to each domain.

# Bandwidth Broker in the Enterprise



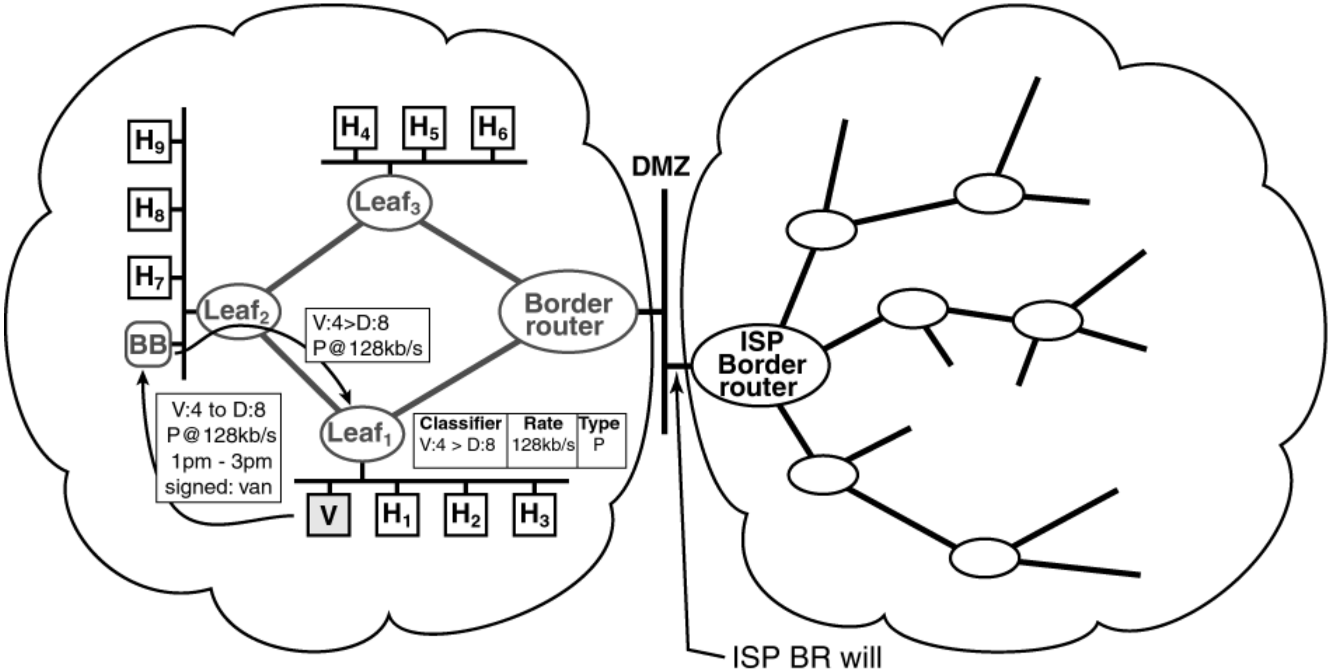
Bandwidth Brokers tell Leaf Routers what to mark based on user requests, current commitments and site policy.

Leaf routers police and mark particular ***local*** traffic based on tspec from BB.

Border routers police marked traffic arriving on interdomain links to negotiated rates.

In a static or configured implementation, BB configures leaf/edges with soft state information (COPS-PR?)

# Requests to a BB and the Result



V:4 to D:8
P@128kb/s
1pm - 3pm
signed: van

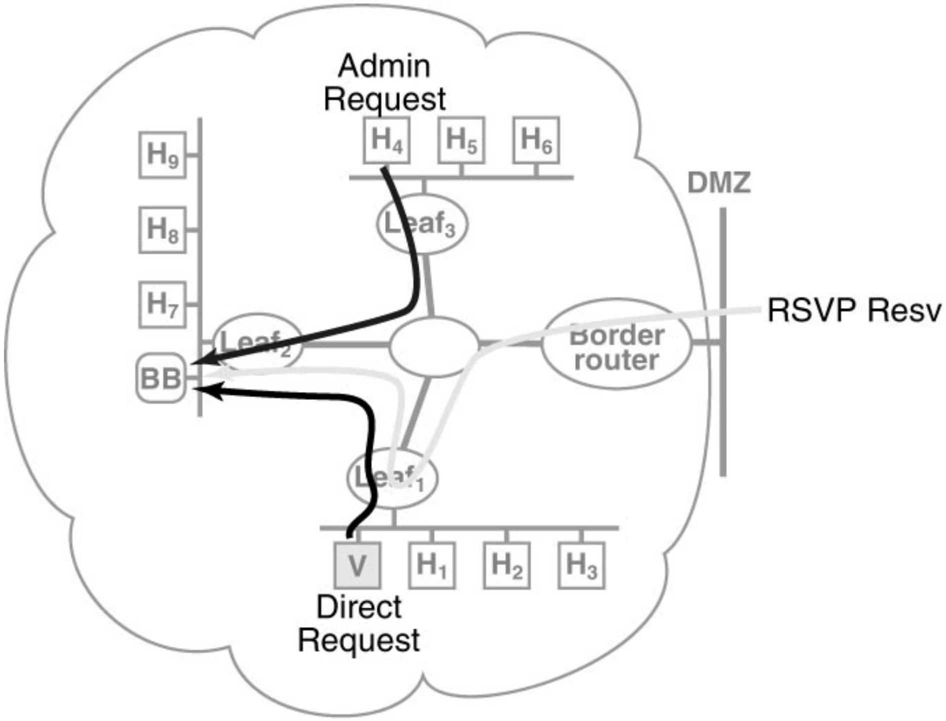
V:4>D:8
P@128kb/s

| Classifier | Rate | Type |
|---|---|---|
| V:4 > D:8 | 128kb/s | P |

ISP BR will
police "P" traffic
arriving on this
link to 512kb/s

Here the BB responds to user requests

# Requests Can Come from Many Sources



“Agnostic about signaling”

# What about Voice?

Inside a high-bandwidth cloud, VoIP flows are trivial and not worth tracking

- › Assume configured EF at each network node sufficient to handle all calls
- › When a call is initiated, check if the destination is within the cloud
- › If so, just admit the call
- › May want to set up a classifer on the edge router to ensure no “spoofing”

# Voice between (Administratively Same) Clouds

Track connections only in the areas of limited resources, boundaries between clouds. For two bandwidth-rich clouds connected by a low-bandwidth link

- › Assume the low-bandwidth link is configured for an EF rate that gives a sufficiently low probability of “busy” (bw_available)
- › When a call is initiated, check destination
- › If it’s in the other cloud, check: (bw_available - call_bw) >=0?
- › If not, refuse call
- › If yes, bw_available -= call_bw and proceed

# Voice across Clouds

Locally, track connections only in the areas of limited resources, tail to “next cloud”.

- Assume the tail is configured for an EF rate that sufficient to handle all outside calls most of the time (bw_available)
- When a call is initiated, check destination
- If it’s “not-me”, check: (bw_available - call_bw) >=0?
- If not, refuse call
- If yes, signal/message “next cloud” and wait for reply
- If reply is positive, bw_available -= call_bw and proceed

# Pointers

- For information on the IETF Differentiated Services Working Group, see www.ietf.org/html.charters/diffserv-charter.html. The diffserv RFCs are at the bottom of the page.
- A talk on diffserv and bandwidth brokers by Van Jacobson given at the Internet2 QoS workshop is available at: www.internet2.edu/media/qos8.ram and the proceedings from that workshop are at www.internet2.edu/qos/may98Workshop/9805-Proceedings.pdf
- A “historical” document that still has relevance about BBs (Nichols, Jacobson, and Zhang): ftp://ftp.ee.lbl.gov/papers/dsarch.pdf