Internet2 QBone—Building a Testbed for Differentiated Services

Authors

Benjamin Teitelbaum (ben@internet2.edu)
Internet2 (UCAID) / Advanced Network & Services

Susan Hares (skh@merit.edu)
Merit Network

Larry Dunn (ldunn@cisco.com)
Cisco Systems

Vishy Narayan (vnarayan@mail.arc.nasa.gov)
NREN/NGI Program, Raytheon/NASA Ames Research Center

Robert Neilson (rneilson@bcit.bc.ca)
British Columbia Institute of Technology

Francis Reichmeyer (franr@iphighway.com)
IPHighway

Abstract

The Internet2 project is a partnership of over 130 U.S. universities, 40 corporations and 30 other organizations. Since its inception, one of the primary technical objectives of Internet2 has been to engineer scalable, interoperable, and administrable interdomain quality of service (QoS) to support an evolving set of new advanced networked applications. Applications like distance learning, remote instrument access and control, advanced scientific visualization, and networked collaboratories will allow universities to fulfill their research and education missions into the future, but only if the network QoS that these applications require can be assured. To meet this challenge, the Internet2 QBone initiative [1] has brought together a dedicated group of U.S. university and federal agency networks, international research networks, engineers, researchers, and applications developers to build a testbed for interdomain IP differentiated services (DiffServ). This paper presents the engineering motivations behind DiffServ and its adoption by Internet2, provides an overview of the QBone architecture, and describes its anticipated deployment, including plans for a trial interdomain bandwidth brokering architecture.

1 Introduction

A primary technical objective of the Internet2 project is to provide advanced IP networking services to support an evolving set of new advanced networked applications. Important advanced applications areas for Internet2 include distance learning, remote instrument access and control, advanced scientific visualization, and networked collaboratories. Beginning in the fall of 1997, the Internet2 QoS Working Group sought to understand the quality of service (QoS) requirements of advanced applications as well as the alternatives for engineering Internet2 networks to meet those requirements. The working group heard from advanced applications developers, campus network planners, and gigaPoP operators, and identified a demanding set of requirements for Internet2 QoS.

Foremost among these requirements is the need to meet the absolute, end-to-end performance requirements of a broad range of advanced applications, including several important applications requiring hard real-time assurances. From Internet2 network engineers and planners came two additional technical requirements: 1) any viable approach must *scale,* allowing core routers to support thousands of QoS-enabled flows at high forwarding rates, and 2) any viable approach must *interoperate,* making it possible to get well-defined interdomain QoS assurances by concatenating the QoS capabilities of several independently configured and administered network clouds.

As the Internet2 QoS Working Group looked for solutions to these problems, the differentiated services (DiffServ) approach to QoS [2,3] began to gain significant interest in the IETF as a lightweight alternative to the RSVP-integrated services (IntServ) architecture [4]. DiffServ is an attempt to design a simple architectural framework for QoS that can provide a variety of scalable, end-to-end services across multiple, separately administered domains, without necessitating complex inter-provider business arrangements or complex behaviors in forwarding equipment. At a workshop in May 1998 [5] the working group presented DiffServ as the architecture best suited to meeting the QoS needs of Internet2, and began a dialogue that culminated in rough consensus around the need to build an interdomain testbed to explore and advance DiffServ.

Although the evolving IETF DiffServ architectural framework offers a promising approach to overcoming the scalability, interoperability, and administrability problems that have plagued previous QoS efforts, the strength of the architecture and the mindshare momentum currently behind it do not guarantee success. DiffServ has not yet been evaluated in the wide-area, and the architectural framework begs many questions and leaves many difficult research, engineering, and policy problems unanswered. For example, it is far from clear how to perform efficient admission control for connectionless networks, what implications DiffServ will have for traffic engineering, how to design protocols for interdomain DiffServ reservation setup, how to provide for reservations in advance (*e.g.* to support scheduled distance learning courses), or what protocols and admission control algorithms are needed to support multicast DiffServ.

The QBone initiative, launched in late 1998, seeks to address these and other problems by building an open and highly-instrumented testbed for interdomain differentiated services. In this testbed, experimental services will be deployed, debugged, analyzed, and refined by networking engineers and researchers working in close collaboration with the users and developers of new advanced networked applications. The QBone initiative will focus initially on deploying and evaluating a single service, the QBone Premium Service (QPS). QPS is based on the expedited forwarding (EF) per-hop behavior (PHB) and described below in the section "QBone Architecture".

Networks participating in the QBone initiative currently include vBNS, Abilene, ESNet, NREN, CA*Net2, SURFnet, TransPac, MREN, NYSERNET, NCNI, and the Texas gigaPoP, as well as numerous universities and labs. A full listing of participants with links to individual QBone project pages may be found from the QBone web site [1].

Engineering, debugging, and understanding new interdomain differentiated services will require tremendous cooperation and openness among all participants. The research and higher education networking community is in general far more open - and interested in finding common solutions - than are commercial Internet service providers. Higher education has a long history of pioneering new internetworking applications, as well as a community of developers eager to begin using and testing new pre-production network services. For these reasons, Internet2 is uniquely situated to build the first interdomain testbed for DiffServ and to begin to tackle the difficult engineering and policy problems that it presents.

This paper describes the motivations behind the QBone, its architecture, and current deployment plans. Section 2 provides an overview of the differentiated services architectural framework, introduces the concept of a *bandwidth broker* (BB) to automate network resource management, and discusses a spectrum of engineering options for achieving resource allocation under the DiffServ framework. Section 3 describes the QBone architecture, including the initial interdomain service to be evaluated and the measurement architecture to be used in that evaluation. Section 4 addresses security considerations in the DiffServ model. Deployment plans are discussed in section 5, with particular emphasis on plans for an experimental evaluation of an inter-BB reservation signaling protocol. The paper concludes with a few forward-looking comments on the QBone initiative, its relevance, and the challenges ahead.

2 Background

2.1 Differentiated Services

For several years, a variety of groups within the Internet Engineering Task Force (IETF) have been working on specifications for mechanisms to provide quality of service (QoS) in IP networks. Much of this effort was focused on the RSVP-integrated services (IntServ) model, which is closely tied to IP multicast and requires each network element in the path from sender to receiver to provide QoS on a per-flow basis. Although IntServ has made significant strides toward providing QoS, it is widely considered too heavyweight to scale across a large internetwork like the Internet. Differentiated services (DiffServ) addresses the issue of scale by defining QoS mechanisms that operate on groups of micro-flows that have similar QoS requirements. The approach is scalable to large networks, and is capable of providing a variety of end-to-end services across multiple, separately administered domains.

The DiffServ architecture [3] achieves its scaling properties by defining a small number of simple differentiated packet forwarding treatments, known as "per-hop behaviors" (PHBs). Individual network elements implement these PHBs through a variety of mechanisms and queuing disciplines. The essence of DiffServ is to combine these differentiated PHBs with carefully configured traffic policing mechanisms at the edge of the network to provide a variety of services. By concentrating policy enforcement activities at the edge and providing simple aggregate data handling in the core, a network operator can ensure that new IP services do not require excessive state information or expensive forwarding decisions in core network routers. Each data packets that enter a DiffServ network is marked with a DiffServ codepoint (DSCP) in a newly defined IP header field (the "DS field") [2] to indicate which PHB should be applied to the packet. Packets marked with the same codepoint are considered a "behavior aggregate" and all receive the same PHB treatment, regardless of the micro-flow to which they belong.

In addition to forwarding engines capable of implementing the emerging PHB standards, the DiffServ architecture requires edge devices that include traffic conditioning components that are able to classify, mark, shape, and drop packets as they enter and leave a DiffServ domain. The first trusted edge router downstream from a sending host performs per-flow policing and marking. As marked packets flow downstream, they are combined with similarly marked DiffServ packets into a behavior aggregate and all subsequent traffic conditioning is performed on aggregated traffic. As conforming traffic traverses a DiffServ domain, it may acquire unacceptable burst characteristics as a side effect of various queuing delays or increased aggregation. In order to ensure that the burst characteristics of aggregated data conform to the traffic specification for the provisioned service, each DiffServ domain must have the ability to perform traffic shaping on each aggregate as it exits the domain, and should shape each aggregate traffic flow as it exits the domain. Failure to shape could result in potentially conforming traffic being rejected by policing mechanisms on ingress to the next downstream DiffServ domain.

At inter-domain boundaries, service level agreements (SLAs) specify the transit service to be given to each aggregate. In general, SLAs are complex, business-related contracts that cover a wide range of issues, including network availability guarantees, payment models, and other legal and business necessities. To facilitate QoS specification within the contract, an SLA will contain a service level specification (SLS) that characterizes aggregate traffic profiles (often based on token bucket parameters) and the PHB to be applied to each aggregate. By carefully enforcing the aggregate traffic contracts between domains and ensuring that new sources of marked packets do not cause traffic profiles to be violated, the DiffServ architecture is capable of providing well-defined end-to-end services over concatenated chains of separately administered domains. Furthermore, since SLSes exist only at the boundaries between domains, DiffServ requires relatively simple bilateral service level agreements that mimic conventional best effort exchange agreements.

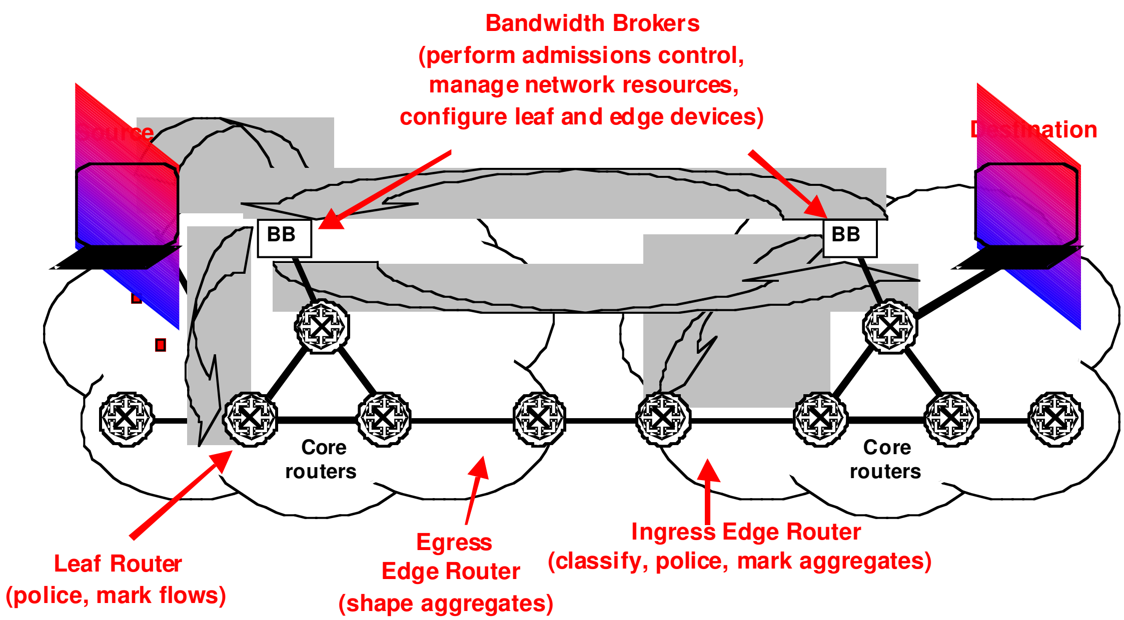

Figure 1. The Differentiated Services Architectural Framework

If the number of different services offered is small and the contracts relatively static, the SLSes between domains may be manually negotiated and the edge devices configured by human network administrators. However, to automate the process of SLS negotiation and admission control, and to configure network devices correctly to support the provisioned QoS services, each DiffServ network may be outfitted with a new component called a "bandwidth broker" (BB). The BB is responsible for ensuring that resources within the DiffServ domain and on links connecting adjacent domains are properly provisioned and not over-subscribed.

2.2 Bandwidth Brokers

The IETF DiffServ framework does not include as a requirement any mechanism for signaling QoS requests between hosts and routers, or between DiffServ-enabled domains. Nor does it include any mechanism for managing the allocation and utilization of DiffServ resources within a domain. These tasks are the responsibility of the bandwidth broker.

A bandwidth broker maintains information relating to the SLSes that are defined between a DiffServ domain and its customers. Customers include local users as well as the adjacent networks that provide connectivity to other parts of the Internet. The BB uses this SLS information to configure the routers in the local DiffServ domain, and to make admission control decisions.

Before marked packets from a data source are admitted to a DiffServ domain, the source must signal its local BB to initiate a service reservation. The potential source is authenticated and subjected to local admission control policies. If the service reservation is admitted locally, the BB may initiate an end-to-end reservation request along the chain of BBs in the DiffServ networks to be traversed by the data flow. When a network-wide admission control decision has been made, the BB will configure the routers in the DiffServ domain to support the requested service profile. The bandwidth broker allows separately-administered DiffServ domains to manage their network resources independently, yet still co-operate with other domains to provide dynamically allocated end-to-end QoS.

The process of establishing end-to-end QoS with bilateral SLAs is best described by means of an example. Consider an agreement between a campus network and its physics department. In this example, in order to run a joint experiment with a remote laboratory, the physics department needs to transmit 10 MB of "Premium" traffic [6] across the Internet. The biology department also has a joint experiment and wishes to send 5 MB of Premium traffic to a remote biology laboratory. To meet these requests, the university network enters into a contract with its Internet service provider (ISP) for 15 MB of Premium service. The ISP might have four other customers requesting 15 MB of Premium service each, so the ISP must negotiate with one of its downstream providers (which could be a large backbone network such as vBNS or Abilene) for permission to send 75 MB of Premium traffic through the network. These service level specifications express the business agreements of the networks, and are used to determine the level of DiffServ provisioning that must be effected in each network if the various service level agreements are to be honored.

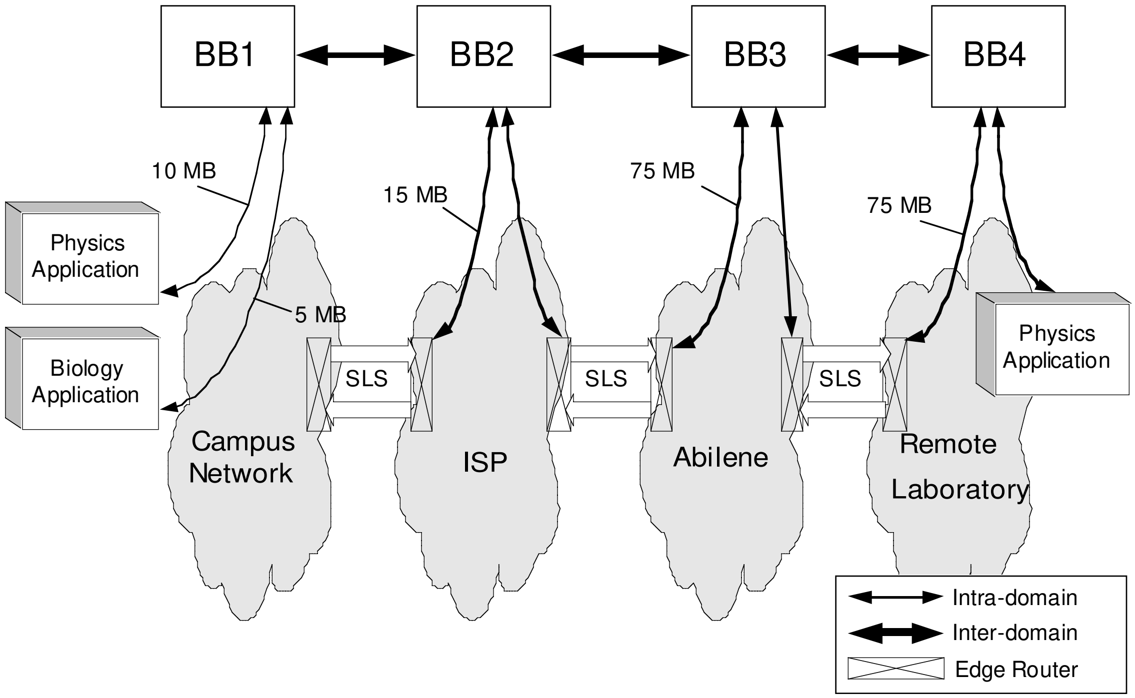

Figure 2. Service Level Agreements on Bandwidth for Academic Institutions

Admission control decisions are based on the usage policies established within a network and between adjacent networks. For example, if the above-mentioned physics department runs an additional experiment that requires a total of 15 MB of service once per week, it should add this requirement to its SLS. The campus network translates this SLS addition into a policy statement that allows one period of 15 MB traffic per week. The physics department's edge router is configured to enforce the standard campus network policy that allows up to 10 MB of Premium traffic. When the physics department wishes to use its once-per-week 15 MB allocation, a request for the additional bandwidth is initiated by the physics application and is sent to the bandwidth broker. The bandwidth broker configures the edge router at the physics department to allow the 15 MB of Premium service. If a second request for 15 MB occurs during the week, the BB will reject the request, and the edge router will not be configured to accept the additional data. Admission control for packets entering a network depends on the partnership between the bandwidth broker's policy decisions and the router's internal mechanisms that enforce those decisions.

As the example demonstrates, the bandwidth broker is required to keep track of QoS resources, make policy decisions based on SLS information, and communicate policy enforcement information to the edge devices within the DiffServ domain. The need to provide these high-level capabilities has motivated the formation of a QBone bandwidth broker working group [7], which is defining detailed requirements for bandwidth brokers [8], as well as a phased experimental deployment to be carried out in the QBone testbed.

2.3 Options for Achieving End-to-End Resource Allocation

The basic DiffServ and bandwidth broker concepts have been described above. Within the DiffServ architecture, there is considerable latitude in how per-hop behaviors (PHBs), bandwidth brokers, and other control-plane mechanisms might be combined to yield useful end-to-end differentiated services. For example, no standard yet exists as to how a DiffServ domain should combine host signaling, PHB configuration, static resource allocation, and BBs to achieve dynamic end-to-end resource allocation. This section describes tradeoffs involved with the various methods under consideration. Subsequent sections provide detail on the particular choices that will be deployed and evaluated in the QBone.

Several possible methods for building DiffServ functionality are enumerated below. Each selects a particular balance among: which device does packet marking, how much signaling information is required, the expected frequency of signaling, and the degree to which resource allocation for a flow or aggregate of flows is recognized end-to-end. The methods are ordered, roughly, from those offering weak assurances in exchange for minimal administrative and control-plane activity, to those offering stronger assurances but requiring more administrative and control-plane overhead.

The methods listed in the spectrum below are bounded on either end. On the minimalist end of the spectrum is the "do nothing" option, which represents current best-effort delivery of packets. On the other end of the spectrum, beyond the DiffServ notion of highly aggregated flows and minimal state stored in the core network elements, lies RSVP-IntServ. Here, resource allocations are signaled via RSVP, and some amount of network state is installed to keep track of and act on commitments for each flow in networks along the flow's entire path.

The spectrum of mechanisms for building and interconnecting DiffServ-capable networks includes:

1. Layer-2 treatment in the campus, static inter-domain bandwidth allocation: Although not quite DiffServ, this idea uses IEEE 802.1p treatment in the campus network to give packets differentiated treatment via layer-2 marking. No explicit DS field marking is done, and no dynamic signaling is required. Some local resource allocation exists (e.g., the buffers in campus layer-2 devices). Links between adjacent DiffServ domains are monitored, and expanded as necessary to give adequate performance. Such a method is discussed further by Terry Gray in [5].

2. Host DS field marking, no signaling: This is a minimalist DiffServ approach. While requiring that a host mark packets with a particular DSCP, the remaining resource provisioning within and between networks is static. Layer-3 devices (routers) might be configured in a variety of static ways: from a single behavior aggregate always being given preferential treatment (to the possible exhaustion of bandwidth for best effort traffic); to configuring a proportion of resources (e.g. output bandwidth) at each layer-3 hop for each behavior aggregate or group of behavior aggregates; to more full-blown metering (measurement), policing (distinct handling of out-of-profile packets), and output link resource (bandwidth) allocation for each behavior aggregate or group of behavior aggregates. Note that this is also a minimalist mechanism in the sense that individual flows are not recognized anywhere in the network, not even at the edges.

3. Host DS field marking, no signaling, some flow-recognition near edge: An extension of item 2 above, this method adds the feature that some form of flow recognition occurs near the edge. Thus manually configured resource commitments might be made, not only to particular behavior aggregates, but also to flow aggregates, or even to particular flows. A flow aggregate might be characterized loosely, by destination network address prefix (e.g., 10 Mbps towards destination=A.B.C.xxx), or might include a source prefix as well (e.g., 10 Mbps between source=E.F.G.xxx and destination=A.B.C.xxx). If a resource commitment is made to an individual flow, the edge (local) network would have to specify and keep track of the full 5-tuple {src-addr, src-port, dest-addr, dest-port, protocol}. The idea is that once a packet is analyzed and handled (at some level of granularity) at the edge, the packet is subsequently treated only as part of a larger aggregate, as indicated by its DSCP. Also note

that if 5-tuples are recognized by the first layer-3 device (router), then one could arrange for that layer-3 device to mark the DS field, rather than requiring the host to do it.

4. Local signaling, static inter-domain provisioning: This introduces the concept that a host or application might dynamically signal for resources. In this simple case, only the local DiffServ domain knows about the dynamic resource requests. A bandwidth broker and policy server might apply administrative policy as to which applications are allowed to generate flows that receive preferential treatment, and dynamically keep track of intra-domain commitments. Layer-3 devices might be reconfigured by the BB as new resource commitments are made. The links across DiffServ domain boundaries are still statically provisioned. Note that this requires careful monitoring of links towards destination domains. Two source domains (for example, AS-1, AS-2) might submit packets in-profile for their individual agreements with a transit domain (say, AS-T). But if both packet streams were destined for a third domain (AS-3), any bandwidth committed towards the destination DiffServ domain might be easily exceeded at the output link from AS-T to AS-3. Protocols under discussion for intra-domain signaling include adaptations of DIAMETER [9] and RSVP [10]. The method most often discussed for a BB to control intra-domain routers is COPS [11], although several home-grown methods, where a BB establishes a telnet session with a router to configure it, also exist.

5. Single-ended signaling, with inter-BB communication: This extends item 4 by keeping the notion that a host or application might express needs to an intra-domain BB, and adding the notion that BBs in different DiffServ domains communicate with each other. The inter-domain communication allows for dynamic adjustment of the commitments made across the boundary between AS-1 and AS-T. Note that the agreement between AS-1 and AS-T is pairwise, but that acceptance of a new allocation level across that link may require AS-T to do some resource re-allocation internally. The inter-domain BB communication protocol must at least include information about resource allocation changes across the {AS-1, AS-T} link. If the BB protocol also includes destination-network information (e.g., that the increase is meant to accommodate extra traffic destined for AS-3), then AS-T receives information that can propagate toward, and adjust the resource allocation across, the {AS-T, AS-3} link. On one hand, this extra information would lead toward more effective allocation of resources, and increase the chance that a packet will actually get preferential treatment end-to-end. On the other hand, this extra information (destination address prefix) is also likely to lead to an increased level of signaling activity in all affected networks. The anticipated relationships among resource allocation quanta, frequency of update, granularity of control, and certainty of commitment are still topics of research. And that's for the unicast case. The multicast case is even more "interesting."

6. Double-ended signaling, inter-BB communication: In [12], a three-part mechanism is proposed. RSVP is used to signal resource requirements in the source DiffServ Domain AS-1. Such RSVP messages are tunneled through intermediate DiffServ domains (AS-T), without elements in AS-T acting on them directly. The RSVP messages, upon arrival at the destination network AS-3, are used by AS-3 to allow more precise destination resource allocation.

In summary, there are many directions that the DiffServ architecture can take as it evolves to provide end-to-end functionality. As described above, there are significant trade-offs among signaling complexity, administrative simplicity, amount of per-flow state, trust, policy expression, strength of assurance, and scalability. Within the QBone testbed, we aim to provide ample room for experimentation to explore this solution space.

3 QBone Architecture

The QBone architecture [13] seeks to build upon the emerging IETF standards for DiffServ forwarding [2,3,14] by specifying an interdomain service, an integrated measurement collection and dissemination architecture, and a set of common operational practices for establishing interdomain reservations. The architecture is expected to evolve incrementally as experience with the QBone testbed is gained.

3.1 QBone Premium Service

The QBone Premium Service will make interdomain, peak-limited bandwidth assurances with virtually no loss, and with virtually no delay or jitter due to queuing effects. The intent of the QPS service is to approximate the "virtual leased line" or "Premium" service proposed by Van Jacobson in [6] and initially demonstrated across ESNet to the show floor of SuperComputing '97 [15].

QPS exploits the Expedited Forwarding (EF) per-hop forwarding behavior [14]. EF requires that the "departure rate of the aggregate's packets from any DiffServ node must equal or exceed a configurable rate" and that the EF traffic "SHOULD receive this rate independent of the intensity of any other traffic attempting to transit the node". EF may be implemented by any of a variety of queuing disciplines, but is best thought of in terms of forwarding EF packets through a strict priority queue. Services like QPS are built from EF through careful conditioning of EF aggregates so that the arrival rate of EF packets at any node is always less than that node's configured minimum departure rate.

A QPS reservation {source, dest, route, startTime, stopTime, peakRate, serviceMTU} is an agreement to provide the transmission assurances of the QBone Premium Service (described below) starting at startTime and ending at endTime across the domain-to-domain chain route between source source and destination dest (these are arbitrary network prefixes) for EF traffic ingressing at source and conforming to the token bucket traffic profile parameterized by:

- token rate equal to peakRate bytes per second;
- bucket depth equal to serviceMTU bytes.

The transmission assurance offered by the QBone Premium Service is as follows:

- Low loss
  This should be very close to zero, but will not be quantified in this service definition.
- Low latency
  Queuing delay will be minimized, but no assumptions regarding minimal latency routing are made.
- Low jitter
  Delay variation due to queuing effects should be no greater than the packet transmission time of a serviceMTU-sized packet at the subscribed rate; no assumptions about jitter due to other effects (e.g. route instability) are made.

Traffic exceeding the profile {peakRate, serviceMTU} should be dropped on ingress to the service endpoint source and not allowed to progress into downstream DS domains. Bilateral QPS reservations have the same structure as wide-area interdomain QPS reservations and have comparable implications for the configuration of ingress traffic conditioners. Note that the service MTU serviceMTU could be larger than the MTU that the user actually intends to use, as it is only intended as a virtual model of the QPS jitter bound consistent with the leased-line nature of the service. Also note that if two domains want to use different serviceMTUs, then reshaping must happen at the boundary if going from a larger to a smaller value.

Consistent with the DiffServ architectural model, all QPS service level specifications (SLSes) are determined bilaterally between adjacent QBone networks (dubbed “DS domains” in [3]). Although the agreements that define SLSes are strictly bilateral, there are technical implications of the QPS described above that impose minimum requirements on QBone SLSes. These requirements, as well as certain recommendations regarding the EF codepoint and the routing of EF traffic, are discussed in the QBone Architecture draft [13].

3.2 Measurement Architecture

To debug, audit, and study QBone services, each participating QBone domain must collect and disseminate a basic set of QoS measurements. The QBone architecture specifies the measurement metrics to be collected at the edges of each participating QBone domain and specifies how these data are to be disseminated to the community. It is expected that the QBone measurement architecture will evolve with the QBone itself, as new measurement metrics and reporting techniques are modified or added with experience.

3.2.1 Measurement Points and Paths

The metrics required of each QBone domain may be obtained through a combination of techniques, including active measurement, passive "sniffing", and SNMP-style polling of network devices. All measurements are to be taken at or as close to interdomain boundary routers as possible. Figure 3 illustrates an example QBone measurement node configuration. Actual configurations will vary significantly according to local issues.

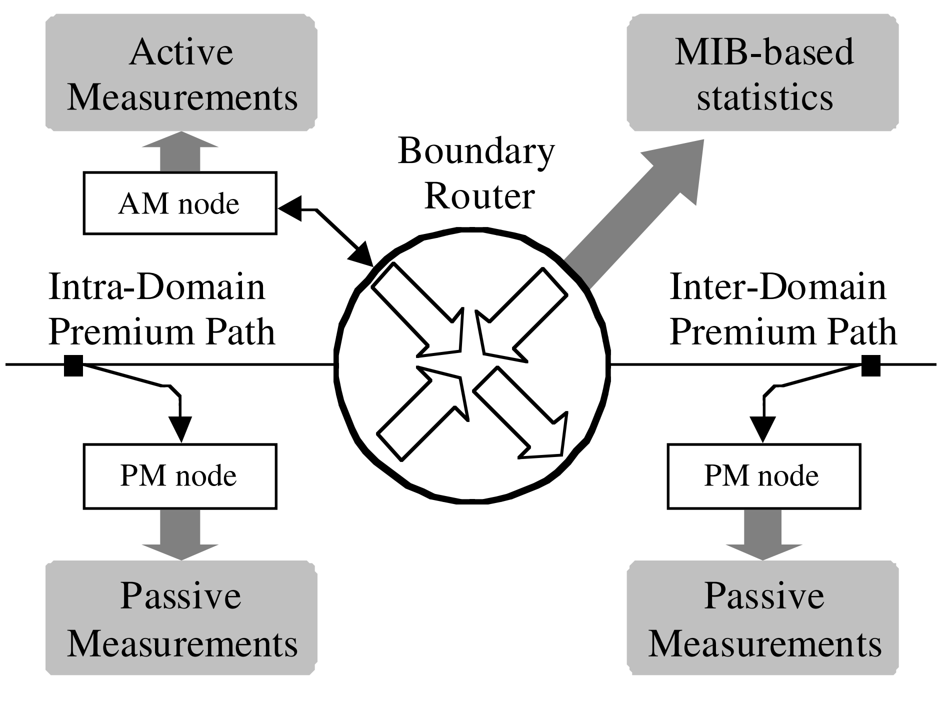

Figure 3. Example QBone Measurement Node Configuration

Each edge router of a QBone domain must be instrumented to serve as a “QBone Measurement Node”. When possible, active measurement probes should have direct interfaces to boundary routers. These probes are the sources and sinks of dedicated measurement paths terminating at the measurement node.

Passive measurement probes must observe the traffic on inter-domain links without perturbing it in any respect. This is commonly accomplished through the use of an optical splitter. Passive measurement probes may additionally be located on intra-domain links to assist in deriving metrics such as expedited-forwarding and best-effort interface discards. Passive measurement equipment may also be used to measure probe flows created by active measurement equipment.

SNMP-style polling agents that extract router statistics to support QBone utilization metrics may be physically located anywhere.

In addition to specifying the metrics that must be collected by each participating QBone domain, the QBone architecture specifies the paths (defined by ordered pairs of measurement nodes) along which path-based metrics, such as one-way packet delay variation, are to be collected.

3.2.2 Required Metrics

The initial design of the QBone measurement architecture is focused on verifying the service goals of the QBone Premium Service and helping with the debugging, provisioning, and performance of EF behavior aggregates. In specifying which metrics are required by the QBone architecture, a concerted attempt was made to strike a balance between an ideal of what would be interesting to measure and a real-world pragmatism that considered only those metrics for which cost-effective implementation techniques were known to exist. Although the QBone architecture makes no specific implementation recommendations for the collection of the required measurement metrics, a companion document is under construction that will provide this guidance to QBone participants.

The required metrics are described below. For each metric, data are to be collected for both the EF and best effort (BE) behavior aggregates.

The first class of required metrics are those that must be obtained through active measurement of paths between QBone boundary nodes. The purpose of these measurements is to build a picture of the network's behavior continuously over time. These measurements will add a small background load to the traffic in each behavior aggregate, but relative to the capacity of the links along each test path, this should be negligible. Small EF aggregates must be reserved for EF active measurement streams, which must take care not to exceed the service parameters of these reservations. Required active metrics for each test path include:

- the IETF IPPM one-way packet loss metric [16];

- an instantaneous one-way packet delay variation metric based on the draft documents [17] and [18];

- periodic traceroutes for each behavior aggregate (formal metrics such as for interdomain path stability could later be derived from these).

The second class of required metrics must be obtained passively - either through sniffing of interdomain links at QBone boundary nodes or through SNMP polling of edge devices for key interface statistics. There is currently only one metric in this class:

- EF and BE load measured in packets per second and bits per second

A final class of required metrics must be obtained through polling of (largely static) provisioning data and through polling of each QBone domain's reservation system. These metrics include:

- link bandwidth in IP bits per second;
- EF commitment in IP bits per second (this is the maximum EF load that a QBone domain would be willing to carry over the given link);
- EF reservation load in IP bits per second (this is the QPS peakRate configured for the given link).

3.2.3 Suggested Metrics

In addition to the required metrics discussed above, the QBone architecture recommends the optional collection of several metrics that are either difficult to collect or are not yet precisely defined. These metrics are suggested because their collection would add significant value to the QBone measurement infrastructure. Currently suggested metrics include: EF and BE interface discards, one-way packet delay, and end-to-end burst throughput tests.

3.2.4 Dissemination Architecture

A key goal of the QBone measurement architecture is to collect, disseminate, and present results in a consistent fashion. This uniformity will greatly simplify the analysis of interdomain QPS reservations and the isolation of any faults in the service. In addition, it will be possible to build a coherent system-wide performance data set that could prove extremely valuable to researchers attempting to model network performance and new experimental DiffServ services.

Each QBone domain must provide a Web site for disseminating and presenting its measurements. Both MRTG-style summary plots and raw measurement data are to be made available through this HTTP interface. The QBone architecture specifies a simple, uniform URL namespace, canonical names for all measurements nodes and metrics, and standard reporting formats for each. From this base, it will be straightforward to create rich tools for auditing, visualizing, or analyzing QBone measurement data.

4 Security Considerations

The primary aim of the DiffServ architecture is to provide different levels of service to different traffic streams on a common network infrastructure. Any techniques used to implement such resource reservations will cause some traffic flows to receive better treatment than others. Two methods of creating a denial-of-service attack are 1) altering the DiffServ field, and 2) injecting packets with the DiffServ field set to a codepoint that make the packets receive enhanced service levels. This theft of resources could result in a denial-of-service attack when the modified or injected traffic depletes the resources available to forward it and other traffic streams.

In a DiffServ domain, any client that wishes to establish a communication channel with a set of guaranteed resources makes a request of a reservation manager. For our discussion, limiting the managed resource to bandwidth only, the resource reservation manager is called a bandwidth broker. A bandwidth broker should be able to provide the requesting client with a reservation for a secure connection either within a single domain or across domain boundaries, for an end-to-end reservation.

4.1 Intra-domain Reservations

When an intra-domain BB receives a reservation request, its first actions are to determine and verify the identity of the requesting client. Depending on the environment in which the BB operates, it may or may not be able to use the environment's authentication mechanism to carry out these tasks. For example, the Globus environment [19] proposes to use the Globus [20] authentication mechanism. All those who exercise authority over the allocation of bandwidth can impose restrictions on its use. Such restrictions, which may constitute the policy governing access to the resource (i.e., the link), could be based on time of day, source address or address prefix, group memberships or application traffic type, or any other form of access control [21].

Once the requesting client has been authenticated, the BB must use a previously-agreed- upon authentication mechanism to determine that all access control policy checks have been satisfied. At this point a successfully established reservation could be represented by a token or an encrypted certificate. All corresponding domain policy checks will entail the BB having access to a certification authority.

4.2 Inter-domain BB and End-to-End Reservations

Security within the inter-domain BBs, like security within other inter-domain protocols such as BGP, has three components: peer identity, link, and data. The BBs in different domains need to establish a bilateral peering (trust) relationship between remote peers. The BB can identify and validate its neighbor by means of authentication tokens, certificates, or pre-configuration. Security of data going across a specific link can be done by using IPSec or other mechanisms which guarantee security across a single link. An end-to-end secure path is set up by establishing secure bilateral peer relationships among the BBs from the source domain to the destination domain.

Data security allows the originator of the resource allocation request (RAR) to secure an individual request. Normally, RARs will be aggregated by the local bandwidth broker. Figure 1 showed the aggregation of the RARs into a single BB to BB exchange. Once the bandwidth broker aggregates the RARs, the combined aggregate sent as an SLS will need to be secured as a new bandwidth request via certificates or authentication tokens. In a few cases, it may be desirable to allow an SLS from the originating domain to be passed intact to the remote domain. For example, in Figure 1, this would be the case if the aggregation of the two RARs into one SLS could be digitally signed and passed through from domain 1 to domain 3.

4.3 Inter-domain Issues for Ingress Routers

An ingress router in a domain is always the first line of defense against any kind of service attacks based on modified codepoints. A node in a DiffServ domain that is the source of traffic acts as a ingress node for that traffic in the domain, and therefore must ensure that all traffic carries acceptable DiffServ codepoints. An ingress router may be required to modify the codepoints of incoming traffic based on previously-agreed-upon service level descriptions. It becomes the responsibility of this ingress node to ensure that incoming packets are in-profile according to the codepoints, and to discard them if they are not. The ingress node may also be required to do traffic conditioning. In addition, the ingress node may need to apply authentication mechanisms to validate some incoming traffic flows, but leave others untouched if the traffic is known to be originating from a trusted source (site) or if the inbound link itself is trusted.

4.4 Interaction of Non-DiffServ with DiffServ Domains

Any links outside the purview of the DiffServ domains and/or the DiffServ network may be subject to local security policies. To ensure link integrity, security on these links may be implemented via physical control devices or by other means such as IPSec. With respect to the use of IPSec within DiffServ domain boundaries, it is worthwhile to note that the IPSec protocol currently requires that the inner header's DiffServ field not be changed by IPSec decapsulation processing at a tunnel egress node. This ensures that an adversary's modifications to the DiffServ field cannot be used to launch theft- or denial-of-service attacks across an IPSec tunnel endpoint, as any such modifications will be discarded at that endpoint. Thus defense against such attacks should consist of a combination of traffic conditioning at DiffServ boundary nodes and the security and integrity of the overall network infrastructure itself.

5 Deployment Plans and Bandwidth Broker Trials

5.1 Initial Deployment

Initial deployment of QoS services will implement host DS field marking, with no signaling, and with some flow-recognition near the edge, as described in section 2.3.

To allow QBone deployment and experimentation to begin as soon as possible, reservations will initially be long-lived and will be established manually, relying on human operators to make admission control decisions, provision appropriately, and configure edge devices. This manual method of reservation will adhere to a set of common operational practices agreed upon by QBone participants. However, it is expected that the complexities of the manual resource allocation, device configuration, and policy management will soon overwhelm the capabilities of a human operator.

5.2 Bandwidth Broker Deployment

In parallel with the initial QBone DiffServ deployment, participants in the QBone bandwidth broker working group [7] are working to define and implement bandwidth brokers, and to agree on a single inter-BB communication mechanism. The introduction of BBs into the various QBone DiffServ domains will be performed in phases, based on the type of admission control functionality provided by the BBs.

5.2.1 Phase 0 - Local Admission

The first BBs deployed in the QBone will operate within single DiffServ domains. Admission control for new resource requests will consider service availability only within the domain where the request is received, and will not validate the availability of DiffServ resources in downstream domains. These phase 0 capabilities require automated resource allocation at the intra-domain level and manual, relatively static, provisioning at inter-domain boundaries.

In this phase BBs will likely be required to configure routers to shape egress flows and police ingress flows, as well as to provide PHBs for QPS and best effort services.

5.2.2 Phase 1 - Informed Admission

Once stable BBs are deployed in a number of DiffServ domains, they will be enhanced to incorporate a simple inter-BB communication protocol. The ability to signal between domains allows BBs to query the availability of resources in downstream domains, resulting in more intelligent admission control decisions.

Resource requests will be aggregated at each inter-BB communication point, and include information regarding the destination of the marked traffic, as well as the resources that are required to support that traffic.

Bandwidth brokers communicate in a peer-wise fashion, using either an end-to-end or an immediate-response signaling model. With the end-to-end model, the BB first determines the availability of local resources, and ensures that the request does not result in the violation of the SLS with the next-hop DiffServ domain. The BB then forwards the request to the appropriate peering BB, which applies its own admission control policies and forwards the request to the next BB. If local admission control fails in any BB along the way, the request is discarded and a negative response indicating where the failure occurred is returned to the originating BB. This process is illustrated in figure 4.

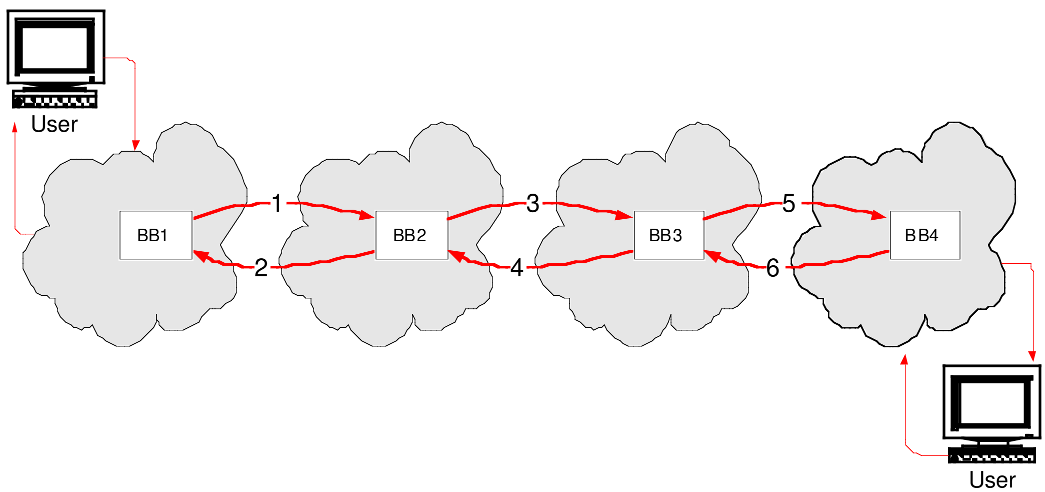

Figure 4. Example Signaling in End-to-End Model

This type of end-to-end signaling is required to establish services which mimic virtual leased lines across DiffServ networks spanning multiple DiffServ domains. However, we are also interested in evaluating other classes of service which do not necessarily require end-to-end resource allocation. An example of this is a service that is deployed only as far downstream as resources permit. This is facilitated by an immediate-response signaling paradigm.

Immediate-response signaling is illustrated in figure 5. Under this scheme the first-hop BB performs local admission control, and forwards the request to the next-hop BB. Downstream BBs apply their admission control policies, and immediately return their responses to the requestor. This allows marked data to begin flowing as soon as possible, and may be sufficient for qualitative-style services that provide better-than-best-effort service, but no explicit QoS guarantees.

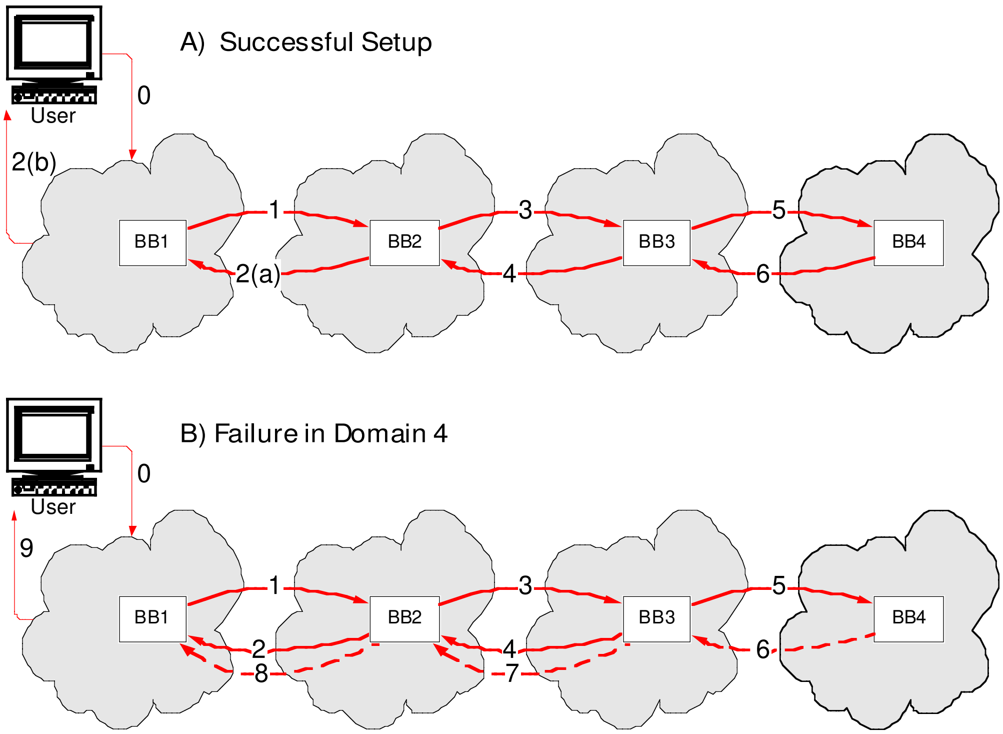

Figure 5. Immediate Response Examples

It should be noted that the immediate response model is not well suited for use with Premium-style services which specify that out-of-profile packets be dropped. For example, consider the situation in which a BB admits a flow from a local user upon receiving a positive response from the neighboring BB, and without any indication of the QoS resources that are available in subsequent downstream DiffServ domains. Traffic could potentially be sent through the first two domains only to be dropped in the third domain due to insufficient QoS resources.

At this point we endorse the deployment of BBs which implement end-to-end signaling, but which don't restrict experimentation with immediate-response signaling in later phases.

5.2.3 Phase 2 - Dynamic Admission

The previous phase implements the reservation of end-to-end resources based on the static provisioning of resources to support SLSes at DiffServ domain boundaries. The objective of phase 2 is to allow for adaptable service provisioning between DiffServ domains. This objective is facilitated by allowing the amount of available service to be adjusted dynamically in response to resource requests.

Adaptable provisioning provides networks with the potential to improve the utilization and allocation of QoS resources within their DiffServ domain.

There are a number of outstanding issues that must be addressed during this phase, and in subsequent deployment phases. These issues include investigation into: the appropriate granularity of SLS adjustment, the need to couple the BB signaling protocol to interdomain routing, the impact of different reservation loads (i.e. large versus small allocations of bandwidth or long versus short reservation hold times), the ability to preempt one reservation for a higher priority reservation, and, finally, techniques to extend DiffServ to multicast flows. These and other issues will be addressed as the QBone and the QBone bandwidth broker experiment progress.

6 Conclusions

The QBone will be the first wide-area test of the evolving differentiated services architecture, and the first experimental deployment of an interdomain DiffServ signaling protocol. It is envisioned that the QBone will grow in scale and robustness as router vendors increasingly incorporate DiffServ functionality in their products and as the engineering of new services to support advanced applications is better understood. By building a highly instrumented testbed that is open and accessible to engineers, researchers, and advanced applications developers, the QBone initiative seeks to advance the state of DiffServ technology and to support the emerging field of DiffServ research. Finally, by working together with the broader Internet2 community to come to terms with the profound administrative, economic, and policy implications of QoS, the QBone aims to start a process that will open the horizon for new advanced networked applications to flourish.

7 Acknowledgements

The authors gratefully acknowledge Guy Almes for his clear-sighted guidance in helping to shape the QBone initiative. Additionally, we would like to recognize Ben Chinowsky for his invaluable editorial assistance and Mariella Wells for her help with the figures.

8 References

[1] *QBone Home Page*, http://www.internet2.edu/qbone/.

[2] K. Nichols, S. Blake, F. Baker, D. Black, *Definition of the differentiated services Field (DS Field) in the IPv4 and IPv6 Headers,* IETF Standards Track RFC 2474, December 1998.

[3] D. Black, S. Blake, M. Carlson, E. Davies, Z. Wang, and W. Weiss, *An Architecture for Differentiated Services*, IETF Informational RFC 2475, December 1998.

[4] R. Braden, D. Clark, and S. Shenker, *Integrated Services in the Internet Architecture: an Overview*, IETF RFC 1633, June, 1994.

[5] *Report from the First Internet2 Joint Applications/Engineering QoS Workshop*, http://www.internet2.edu/qos/may98Workshop/9805-Proceedings.pdf, May 1998.

[6] K. Nichols, V. Jacobson, and L. Zhang, *A Two-bit Differentiated Services Architecture for the Internet*, ftp://ftp.ee.lbl.gov/papers/dsarch.pdf, November, 1997.

[7] *QBone Bandwidth Broker Advisory Council Home Page*, http://www.merit.edu/working.groups/i2-qbone-bb/.

[8] R. Neilson, J. Wheeler, F. Reichmeyer, S. Hares, *A Discussion of Bandwidth broker Requirements for Internet2 QBone Deployment*, version 0.3, http://www.merit.edu/i2-qbone-bb/doc/BB_Requirements_v3.doc.

[9] P. Calhoun, G. Zorn, P. Pan, *DIAMETER Framework Document,* Internet Draft, December 1998.

[10] R. Braden (Editor), L. Zhang, S. Berson, S. Herzog, S. Jamin, *Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification,* IETF Standards Track RFC 2205, September 1997.

[11] F. Reichmeyer, H.K. Chan, D. Durham, R. Yavatkar, S. Gai, K. McCloghrie, S. Herzog, A. Smith, "COPS Usage for Policy Provisioning", Internet Draft, February 1999.

[12] Y. Bernet, R. Yavatkar, P. Ford, F. Baker, L. Zhang, M. Speer, R. Braden, *Interoperation of RSVP/Intserv and Diffserv Networks,* Internet Draft, March, 1999.

[13] *Draft QBone Architecture*, http://www.internet2.edu/qos/wg/papers/draft-i2-qbone-arch-06.html, January, 1999.

[14] V. Jacobson, K. Nichols, K. Poduri, *An Expedited Fowarding PHB,* IETF Proposed Standard RFC 2598, June 1999.

[15] R. Nitzan, *Experience with a Class Based Queuing Demonstration*, http://www.es.net/nesg/esnet-qbone-participation/cbq_test_paper.html, February, 1998.

[16] G. Almes, S. Kalindi, M. Zekauskas, *A One-way Packet Loss Metric for IPPM* , IETF Internet Draft, February 1999.

[17] C. Demichelis, *Instantaneous Packet Delay Variation Metric for IPPM*, Internet Draft, November 1998.

[18] *ITU-T Recommendation I.380: Internet Protocol Data Communications Service - IP Packet Transfer and Availability Performance Parameters*, ITU-T SG 13, February 1999.

[19] I. Foster, C. Kesselman, Globus: *A Metacomputing Infrastructure Toolkit*, Intl J. Supercomputer Applications, 11(2):115-128, 1997.

[20] I. Foster, C. Kesselman, Gene Tsudik, Steven Tuecke, *A Security Architecture for Computational Grids*, Proc. 5th ACM Conference on Computer and Communications Security Conference, pp. 83-92, 1998.

[21] G. Hoo, W. Johnston, I. Foster and A. Roy, *QoS as Middleware: Bandwidth brokering System Design*, Paper submitted for the Eighth International Symposium on High Performance Distributed Computing.

[22] O. Schelén, S. Pink, *Resource Reservation Agents in the Internet*, Proceedings of 8th International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'98), Cambridge, United Kingdom, July 1998.