Quality of service overview

2.1 Introduction to QoS

This section provides a brief description of the main techniques concerned with engineering support for QoS in packet networks. A more detailed description of QoS can be found in a companion document [QoSBasics].

As introduced in section 1.3, QoS is a technology that manages network resources to reduce the negative effects of congestion in packet-switched networks. QoS aims to keep specific characteristics of a network transport service such as packet delay, jitter and loss within desirable limits. All three of these will adversely affect voice, video and other real-time applications; delay and loss may also be perceptible when browsing web pages and loss will affect the bulk transfer rate of data. As traffic in packet networks is random in character, all three QoS characteristics are of a statistical nature. QoS characteristics belong to a wider set of network performance characteristics which along with latency/loss metrics usually include, for example, transport service availability, nodes reachability, packets re-ordering etc.

To keep latency (delay and jitter) and loss characteristics of traffic within desirable limits, QoS re-allocates or provisions available bandwidth between traffic of different applications (or application classes) according to need. To implement this QoS provisioning, network parameters can be tuned to optimise network behaviour so that the necessary amount of bandwidth is provisioned to different traffic classes or individual traffic flows. This optimisation is implemented through sets of queues within network nodes with different types of scheduling and traffic conditioning mechanisms. To differentiate QoS from other techniques of network optimisation it should be noted that QoS (in its general meaning) does not use network parameters such as traffic paths (used in traffic engineering) or link capacity (used in network engineering).

2.1.1 QoS Concepts and Elements

QoS deployments involve a combination of the following components:

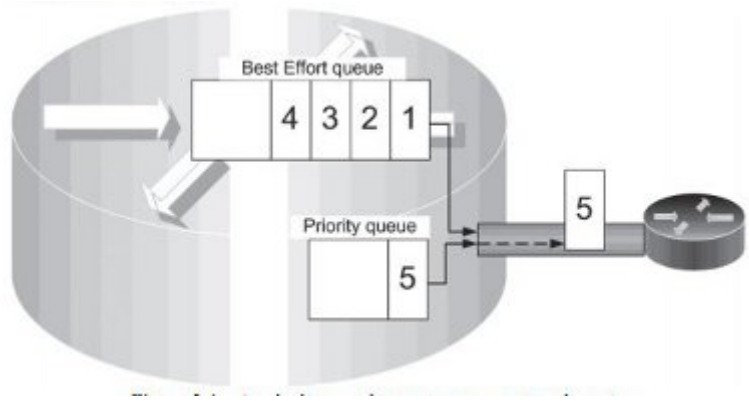
- packet scheduling (queuing)
- traffic classification
- traffic policing and shaping
- active queue management
- resource reservation/provisioning and admission control.

There are different models and implementations of QoS which use some or all of these QoS components in specific ways. However, the use of these components is not exclusive to QoS and they may be used in other areas and for other purposes, for example, packet classification by security systems.

2.1.1.1 Queuing

The main principle of QoS is the existence of several queues that are treated differently by a router/switch, instead of the one queue for all traffic. Some of these queues receive preferential treatment over the default Best Effort queue, providing so-called elevated QoS services. It is also possible to configure queues to receive a lower priority than the default priority. Lower than best effort treatment is known as a Less than Best Effort (LBE) service.

Preferential queues may be based on priority or are rate-based (also known as weighted queuing), or some combination of the two. Network industry best current practice usually consists of configuring at least one priority queue for more time-sensitive traffic, in addition to the default best effort queue. Figure 2-1 shows how traffic arriving on the priority queue (5) might be transmitted before best effort traffic (1,2,3,4) even if it arrives later. Other ratebased queues for traffic that is not so tolerant, yet still sensitive to delay (i.e. non-elastic), might also be configured.



uter

[1]

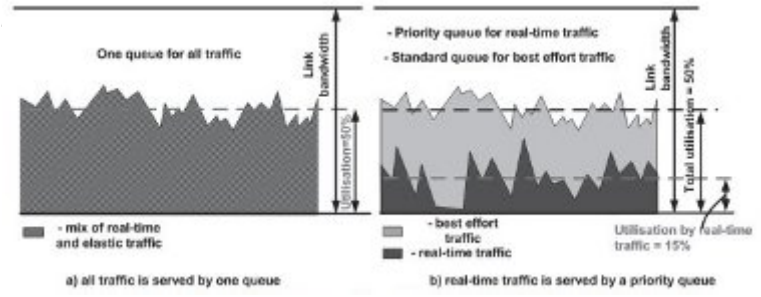
In such architectures, first the priority queue is emptied by the router scheduling process and then packets in the other queues are processed. The priority queue always has a positive effect for packets submitted to it, as they are treated preferentially, in isolation from other traffic. Hence there is a lower level of delay as shown in Figure 2-2.

Rate-based queues are emptied by the scheduler at a specified rate which is determined by the ‘weight’ configured for each queue. Rate-based queues are configured to receive a guaranteed minimum percentage of bandwidth during periods of congestion. It must be stressed that a rate-based queue itself may or may not provide preferential treatment. The critical point here is the ratio of traffic rate going through the queue compared to the amount of bandwidth allocated to that queue which may be called the ‘relative utilisation’. If this ratio is low enough, say less than 20%, then the treatment will be preferential whereas if the ratio is high, for example 60%, then the treatment will be rather poor.

Of course, the existence of several queues with different types of scheduling (de-queuing)

does not solve the problem as a whole. Other QoS elements are necessary to achieve desirable balance between the needs of different traffic kinds and available bandwidth. As the excessive use of a priority queue might lead to the starvation of all other queues, a variety of mechanisms are used to guard against this – the most common being access control and policing.

Figure 2-2 - Effect of a priority queue: lower utilisation for priority traffic a variety of mechanisms are used to guard against this - the most common being access control and



[2]

2.1.1.2 Classification

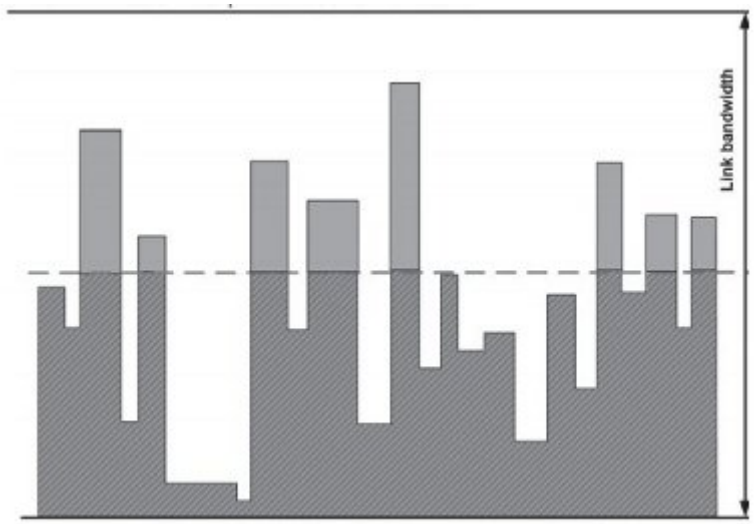
One component of QoS is classification of packets with the ultimate goal of determining which output queue they are placed in. In other words, a router performs access control on incoming packets for its set of established queues. Such classification might use different packet attributes for identification and, in some cases, authentication of traffic to make a selection decision. Classification might be implemented only on the router transmitting the packet into the network; however if that router is not in the same administrative domain as the rest of the network then it may be that the ingress router in the next domain needs to re-classify the traffic. Packet classification in routers is often implemented by means of access control mechanisms such as access lists or filters.

2.1.1.3 Policing

Policing is an enforcement technique used to ensure that the bounds set for each queue are not exceeded. Policing is used to check that traffic entering a network does not exceed the limits configured for its class - for example, to ensure that the maximum percentage of bandwidth assigned to a class is not exceeded. Packets that exceed the configured bounds are either dropped or re-classified to be placed in another queue depending on the policy of the administrator.

Applying classification and policing to a priority queue makes it possible to assign it a maximum percentage of the bandwidth, which will not be exceeded as shown in Figure 2-3. This protects the remaining traffic on the network from being mistreated, which in an extreme case could be complete starvation of bandwidth.

Figure 2-3 - Policing effect: dropping excess of traffic



[3]

2.1.1.4 Traffic Shaping

Traffic shaping is the process of delaying packets within a traffic stream to keep its rate within specified limits. This is similar to policing as shaping is also based on controlling the traffic rate but differs in terms of reaction to non-conformant packets, delaying them instead of dropping or re-marking. As shown in Figure 2-4, traffic that exceeds a pre-defined threshold is held until utilisation drops again and it can be transmitted. This has the aggregate effect of smoothing the overall transmission rate by cutting the peaks and filling the troughs. In addition to this, while policing is applied to the ingress stream before scheduling, shaping is applied to an egress stream after scheduling. The aim of such an operation is to make egress traffic conform to some agreed rate limit which might be policed further by downstream routers and dropped if it is not shaped.

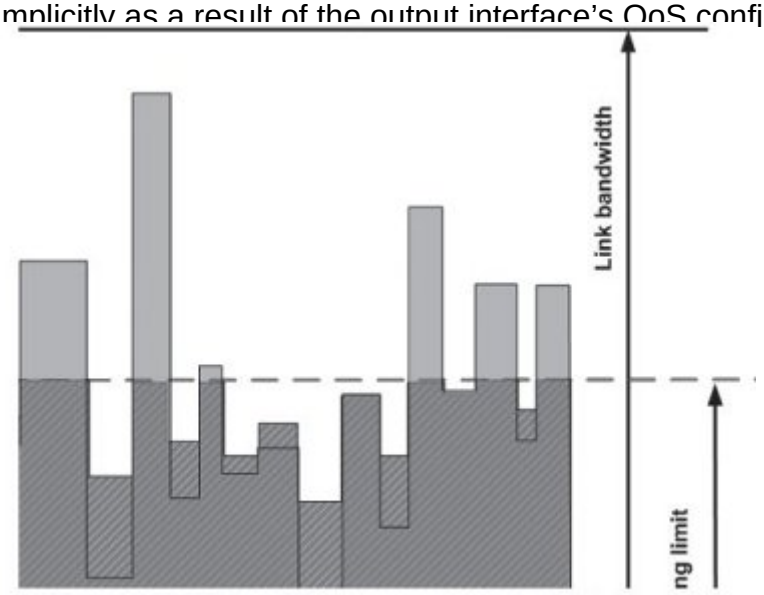
2.1.1.5 Active Queue Management

Active queue management is the proactive approach of indicating incipient congestion to the sender *before* a buffer overflow happens so that senders are informed early on and can react accordingly. In other words this technique uses feedback to avoid congestion. A classic and widely used example AQM technique is the Random Early Detection (RED) mechanism. RED exploits a TCP feature of slowing down the rate of packet generation by a source node in the presence of packet loss. RED drops packets randomly depending on the average queue size. Weighted RED (WRED) is a mechanism that is used to support QoS in a system with more than one queue, whereby WRED applies different dropping policies to each queue.

2.1.1.6 Resource (bandwidth) Reservation/provisioning

QoS techniques at the packet level focus on how to deal with contention for resources, the most obvious of which is bandwidth on network links. Contention for internal resources in switches and routers may be another factor, as in addition to the buffering/queuing on link interfaces, there may also be buffering internal to a network device itself. For example, despite many queues being available on the link interfaces, there may be a limited number of queues (or other capacity limitation issues) across the device’s backplane or switching fabric. As such, an understanding of the internal architecture of each device type in a network may prove useful when attempting to troubleshoot QoS related problems. However, the bandwidth

allocation and queue configuration of an output port are usually the only type of resources which are explicitly specified when QoS is configured on a router or a switch. This does not mean that other router or switch resources, for example processor power, switch fabric or share of the scheduler cycle time, are not reserved. These resources are simply reserved implicitly as a result of the output interface's QoS configuration.



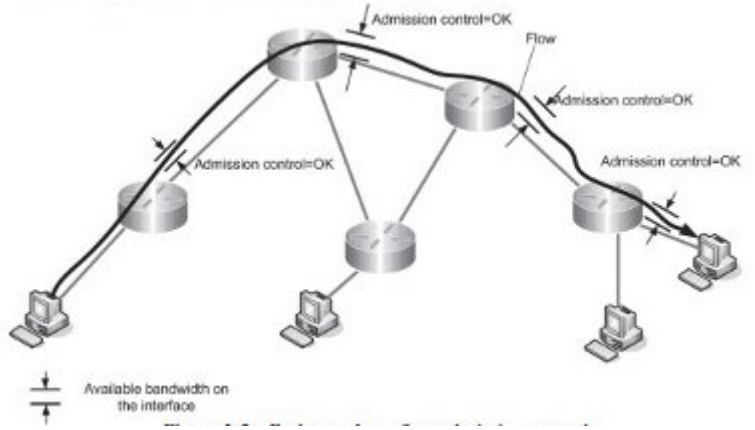
[4]

The general terms 'resource' or 'resources' are therefore often used interchangeably with 'bandwidth' in practice. Bandwidth reservation/provisioning in packet-switched networks can be an important tool and it is significantly different from that in circuit-switched networks. In circuit-switched networks, a fixed part of the bandwidth (a circuit) is reserved for a particular flow on an end-to-end basis. This is a hard reservation in that even if the flow is not using the entire capacity of the circuit, the unused bandwidth cannot be used by other traffic. Another feature of bandwidth provisioning in circuit-switched networks is that it is embedded in the technology, i.e. a circuit establishment just works in this way so that a bandwidth allocation to a circuit happens by design.

Some applications are well suited to circuit-switched networks - for example, a typical telephone call uses 64Kbit/s of bandwidth. If a dedicated voice network provides switched 64Kbit/s circuits, each will be fully utilised - although this may not be a particularly efficient use of the total available bandwidth if many 64Kbit/s circuits remain unused. Congestion on a circuit-switched network is detected before any traffic actually flows - if the resource to create the circuit is not available, the network will deny access.

Bandwidth provisioning in a packet-switched network is similar to that in circuit-switched networks in that it is also concerned with allocating a certain amount of bandwidth for a certain type of traffic. However, in packet-switched networks it is much more flexible - if the reservation is not fully utilised, the bandwidth is available for use by other traffic. This permits more efficient use of bandwidth in the network but introduces the concept of gradual service degradation due to congestion, rather than the flat refusal of service when bandwidth in a circuit-switched network is fully allocated. Another distinction of resource reservation in packet-switched networks is that it may be done both on a per-flow end-toend basis (simulating the behaviour of circuit-switched networks) or on a per-hop basis.

The former case includes an operation called **admission control** where a request for bandwidth is checked against available bandwidth along the possible route of traffic. In fact, it is a flow admission control; a network agrees to serve a flow with a specified rate as there is enough bandwidth to do so (Figure 2-5). If admission control for a requested bandwidth reservation is positive, the network devices keep track of this reservation in the form of QoS configuration and serve traffic accordingly.



[5]

When bandwidth provisioning occurs on a per-hop basis, no flows or their paths are taken into account. This kind of reservation provisions bandwidth on the router interfaces for traffic aggregates passing through the router. The traffic classes consist of aggregate flows of traffic which have similar requirements in terms of delay and loss. In this case, the term ‘admission control’ is not applicable in its traditional sense as an end-to-end per-flow check of resources. However, there are some RFCs, e.g. RFC 4594 [RFC4594] (Configuration Guidelines for DiffServ Service Classes) and RFC 3798 [RFC3798] (A Framework for Integrated Services Operation over DiffServ Networks), and vendors’ documents that use the term ‘admission control’ as a mechanism which controls and limits the rate of particular ingress traffic, i.e. policing. In this document, admission control always refers to its traditional meaning - checking availability of bandwidth to carry a requested load along the traffic path on an end-to end basis.

Proper bandwidth provisioning plays an important role in providing QoS as an incorrect bandwidth allocation to some flows may result in worsening the transport service, rather than improving it. Therefore, the aim of bandwidth provisioning is to configure routers or switches correctly in order to support the bandwidth allocation decisions made. Bandwidth provisioning in QoS involves several distinct stages:

- evaluating traffic requirements and rates. This may involve some monitoring of network and applications behaviour
- allocating available bandwidth to different traffic classes or flows
- implementing bandwidth allocations by means of the available QoS mechanisms: classifiers, queues, policers etc.

Bandwidth provisioning plays an important role in providing consistent, robust Quality of Service. However, this is not exclusive to QoS as bandwidth provisioning is also an important measure in plain networks where no QoS mechanisms are deployed.

2.1.2 QoS Models

QoS mechanisms should be used in a systematic manner, i.e. according to a particular model. Two such QoS models have been standardised by the IETF for IP networks.

The first model is known as DiffServ [RFC2475], where the network resources that are reserved take the form of interface buffer space/queues and a percentage of the link bandwidth is assigned for each type of traffic. This configuration is required on each QoSenabled interface on the network and there is no interaction with any other node, i.e. packets are treated on a per-hop basis (PHB). Further details are given in section 2.1.3.

The alternative model, IntServ [RFC1633], is based on the end-to-end reservation of an amount of bandwidth for an individual flow between two points. The routers along the path conduct a negotiation to confirm that the bandwidth is available and mark it as reserved, and each router keeps track of that reservation. Unlike DiffServ, this guarantees the end-to-end bandwidth. Further details are given in section 2.1.4.

2.1.3 DiffServ

The Differentiated Services (DiffServ) [RFC2475] approach involves the reservation of network resources such as output interface buffer space/queues and percentages of link bandwidth assigned for each type of traffic. Traffic types (which may also be called RFSs - Resource-Facing Services) aggregate flows with similar delay and packet loss requirements. DiffServ in itself makes no absolute guarantees other than that different traffic types will be treated in different ways, according to the QoS parameters configured.

This configuration is required on each QoS-enabled interface in the network and there is no interaction between nodes so packets are treated on a per-hop basis (PHB). This means that each router on the path between the source and destination may have different QoS parameters configured. For example, the amount of bandwidth assigned for a traffic type on a backbone network link is likely to be greater than that on an access link, as the backbone may be carrying many flows of that type, from different sources.

Varying the bandwidth parameter is usual on any network with a backbone and many access links, and is unlikely to be harmful. However, varying other parameters may result in traffic receiving priority treatment in one router and a different treatment in the next, so a consistent approach to setting QoS parameters across a network is important. For this reason when two different networks, for example JANET and GÉANT, attempt to interwork using DiffServ, it is vital that both parties understand the other's QoS architecture. To simplify inter-domain configuration, the IETF recommended two main types of PHB: Expedited Forwarding (EF) [RFC3246] and Assured Forwarding (AF) [RFC2597]. EF assumes the best quality of treatment in terms of latency/lost parameters which a router can provide to packets. AF is mostly designed for traffic which needs guaranteed delivery but is more tolerant to packet delays/loss than traffic which requires EF. Neither EF nor AF definitions specify any particular details of router configurations such as queuing, admission control, policing and shaping types and parameters to implementation.

DiffServ is a stateless architecture in that a packet enters a router, is classified as necessary, and then placed in the appropriate queue on the output interface. The router does not attempt to track flows and, once a packet has been transmitted, it is forgotten.

According to the DiffServ approach, any network router can carry out traffic classification, i.e. decide what PHB should be applied to arriving packets, independently. However, DiffServ defines a special field in the IP packet called DSCP (Differential Services Code Point) which can be used as an attribute indicating the desirable PHB for this packet. The DSCP field is usually intended to be used within a network where routers trust each other, as in a single administrative domain. In such a case, only the edge routers of the network perform classification and mark ingress packets with a specific DSCP value; all the core routers can then trust this choice and treat packets accordingly. By extension, DSCP values can also be used as a means to coordinate traffic handling between trusting networks such as JANET and Regional Networks. The use of the DSCP field is not mandatory; it is a tool for loose coordination of a network of routers which is intended to decrease the amount of packet processing work for the core routers.

2.1.4 IntServ and RSVP

Integrated Services [RFC1633] [RFC3936] is the other IETF approach to QoS, and is based on the end-to-end reservation of an amount of bandwidth between two points. The routers along the path conduct a negotiation to confirm that the bandwidth is available and then mark it as reserved, each keeping track of the reservation. The negotiation process is known as the Resource Reservation Protocol, or RSVP.

Unlike DiffServ, this will statistically guarantee the end-to-end bandwidth but is less equipped to control delay, as it does not necessarily imply the use of multiple or prioritised output queues. IntServ is stateful, i.e. each router must maintain information about each IntServ flow it has accepted, until the removal of its reservation. Where a network carries many flows using IntServ, this may lead to significant extra processing and memory resources being consumed by the setup and teardown of reservations, and the state information that must be kept per reservation. On a backbone router within a large network, there may already be high demands on such resources, particularly in terms of memory if large IP routing tables are involved.

For this reason, IntServ is most commonly deployed for a few very long lasting flows, such as long term traffic-engineered paths which may remain in place for weeks, months or longer.

2.1.5 Comparing QoS Models

Both models, IntServ and DiffServ have their benefits and drawbacks.

IntServ can provide QoS with strong guarantees for each flow crossing the network on an end-to end basis (sometimes called ‘hard QoS’). However, it is not scalable as the state information is proportional to the number of flows and this is unacceptable for backbone networks. Another feature which makes IntServ less attractive to providers is that it allows end user applications too much freedom to initiate resource reservations. The more familiar operational model for providers assumes that they are in full control of all network resources, and are not subject to external or ‘foreign’ RSVP requests into their network.

As such, the biggest benefit of DiffServ over IntServ is that it removes the scalability issue at the expense of deterioration of QoS. DiffServ QoS cannot support the strict QoS guarantees associated with IntServ, and this is why it is sometimes known as ‘soft QoS’. The main problem of the DiffServ approach is the uncertainty in traffic class compositions and associated data rates, which results in uncertainty of overall QoS characteristics.

It has been suggested that combining both approaches might produce good results. For example, IntServ might work on the periphery of a network where the number of flows which need reservation is relatively small. DiffServ might then be the more appropriate solution for higher level networks and backbones if they (or some of their parts) are not adequately provisioned. Another way of improving the performance of DiffServ may involve the tracking of individual flows using systems other than the routers themselves, e.g. manual tracking or tracking by external software systems. Manual tracking is likely to be very time and resource intensive and hence error-prone, but this may depend on actual network topology. The use of external software systems for improving DiffServ performance is known as **bandwidth brokering**, but unfortunately this approach is completely non-standardised at the moment. It is being investigated in some research projects.

2.2 Associated Technologies

The basic QoS models discussed above can be applied to IP networks to assist in the (de)prioritisation of particular traffic based on certain criteria by configuring router devices within the network. However, simply applying the base IP QoS methods alone may not be sufficient on today's typical IP networks. While packets may be prioritised and classified as desired within the IP networks, there are other network elements and protocols that must be considered. We discuss these in general in the following sub-sections.

2.2.1 Firewalls and Middleboxes

The original Internet architecture consisted of hosts and routers, with routers performing best effort forwarding of packets towards their final destination. Over the years, a number of ‘middlebox’ appliances have been designed and deployed that have the side effect of interfering with this end-to-end principle. There are now many examples of such middlebox devices, for example:

- firewalls or content filters
- caches
- proxies.

In each case, the device performs network or application oriented services and may often improve the performance or capability of an application. However, if these middleboxes do not support the base QoS functions (e.g. RSVP or DiffServ classes) then they add an additional hop (which may or may not be transparent to a user) at which the desired QoS behaviour may not be realised.

Further, even where QoS functions are in principle supported, it may be that the regular processing of that device (e.g. a heavily loaded firewall) adds undesirable latency or jitter to a service or data stream. Thus, care needs to be taken when such devices are on a QoS path. Given that such devices tend to be deployed at, or near, a site border, site administrators should check QoS capabilities of such devices when specifying them in scenarios where QoS methods are in use. 2.2.2 Network Address Translators (NATs)

A NAT is a middlebox that warrants special consideration. A NAT device typically embeds two capabilities; an address (and often port) mapping function between public and private IP address ranges used on each side of the NAT, and usually a number of Application Layer Gateway (ALG) functions, e.g. to enable FTP to operate through a NAT where the NATed IP address is included in the FTP data payload.

NATs add complexity to QoS deployment because of their address translation property. If a QoS path is required between a specific internal and external node or network, configurations need to be aware of that. For example, if a specific IP videoconferencing unit has a private address, its traffic needs to be mapped to a public IP address for external communication where appropriate traffic from that public address is given due QoS treatment. It may be that the QoS classification is done at or before the NAT device, in which case the DiffServ value needs to be retained during header translation.

While QoS can be deployed between NATed and public IP networks, it is simpler to configure and monitor/manage when public IP addresses are in use throughout the path.

2.2.3 Virtual Private Networks (VPNs)

A VPN is another ‘special’ middlebox, in that it acts as a network concentrator for what is essentially encapsulated/tunnelled traffic. When a user connects to a VPN service, their traffic is routed via their home VPN server to its destination. While there is usually an option within VPN clients to tunnel traffic only to the home network via the VPN server (and allow the rest to travel directly), the VPN server will add latency (invariably a longer path, even though the hop count to the user appears less) and if heavily loaded may incur packet loss or induce significant jitter.

A site’s VPN service needs to be adequately provisioned to ensure that its users do not suffer a degraded service in comparison to using a non-VPN connection. Even then, it is unlikely that an IntServ or DiffServ QoS solution can be facilitated where VPNs are used unless the provisioning can be made on each hop the VPN traffic traverses, and the VPN server system supports the QoS method.

2.2.4 Ethernet LANs

The IntServ and DiffServ methods are generally applied on routers. These technologies cannot impact QoS within a LAN. As such, if there is congestion within a LAN, or a requirement to prioritise certain traffic on a LAN, e.g. to a specific Ethernet switch port, then other solutions are required. In particular the IEEE 802.1p standard offers a method to prioritise Ethernet (MAC) frames in a LAN. This may be useful for example where an IP videoconferencing unit or VoIP (Voice over IP) devices share a LAN with general PC or laptop devices.

2.2.5 Wireless LANs

Wireless LANs require particular consideration because, unlike modern Ethernet LANs that are switched (bridged) per port, a wireless LAN is generally a shared medium between nodes associated to the access point. In addition, the bandwidth available is currently very limited (11/54Mbit/s) in contrast to wired Ethernet LANs (100/1000Mbit/s). Methods for providing QoS on wireless access points (APs) appear to be somewhat vendor specific at present. There is support for QoS based on IEEE 802.1p or 802.1q (VLAN) tags on Ethernet frames. However, the wireless interface itself is also subject to interference from static or moving objects, and as such an absolute QoS guarantee on wireless LANs is difficult to provision.

2.2.6 Multiprotocol Label Switching (MPLS)

MPLS is a technology that marries IP networks with the connection-oriented virtual circuit technique which was initially introduced in X.25 networks and then used in frame relay and ATM networks. MPLS uses the IP control plane (routing protocols) and frame relay style of data forwarding (layer 2 functionality). MPLS functionality is usually implemented as an additional feature in router software which may be or may be not activated by a network administrator.

MPLS was initially developed for speeding up the forwarding of IP packets by routers but afterwards several other applications of this technology arose; the most important of them are MPLS VPN and MPLS Traffic Engineering (TE). It is also important to mention that routers can combine standard IP hop-by-hop forwarding for some of the traffic with MPLS forwarding

for others.

MPLS itself doesn't provide QoS. However, it can be termed as a QoS supportive technology because of its TE capability. MPLS TE (as any other transport technology with TE functionality) provides a good foundation for guaranteed QoS as it gives full control over traffic routes. Such a control means that it is possible in principle to check available capacity along deterministic QoS traffic routes and admit traffic only if sufficient capacity is available.

MPLS has not been deployed on the JANET core and has found only very limited use across Regional Networks. However, MPLS is popular among commercial providers for VPN purposes.

2.2.7 Multicast

IP multicast is in itself a form of QoS solution in that it can conserve considerable bandwidth in certain scenarios, leaving the remaining bandwidth less contended for regular unicast applications. That said, in some cases, QoS is desirable for multicast traffic, e.g. an important live video streaming of some event or seminar.

There are many issues to consider when deploying QoS for multicast itself. Perhaps the hardest is the requirement to provision or reserve bandwidth for multicast traffic, where knowing the locations of all senders or receivers in advance of a multicast session being run is not a trivial task. This problem is discussed in RFC 3754 [RFC3754] in some detail. Other considerations for Multicast QoS include:

- handling of initial ASM traffic routed via a Rendezvous Point (RP); encapsulation adds latency, as may the path via the RP until the optimal Shortest Path Tree is established
- tunnelled IP-in-IP traffic, often used in the multicast 'mbone', to bypass non-multicast capable routers. Provisioning QoS for such tunnels adds complexity
- presence or lack of IGMP or MLD snooping on layer 2 devices (which otherwise may lead to undesired multicast traffic flooding) - this is perhaps more an issue of protecting non-multicast traffic
- capability of host operating systems/hardware to process multicast; in some cases, multicast handling may be poor.

Multicast is a very valuable tool for bandwidth conservation but operational examples of delivering QoS for multicast traffic are still quite rare.

2.2.8 IPv6

IPv6 deployment will continue to grow as the IPv4 public address space approaches exhaustion (or at least the point at which getting public IPv4 address space becomes very hard). There are two particularly interesting aspects to IPv6 and QoS. The first is whether IPv6 offers any new QoS solutions. While DiffServ remains the same (8 bits for the Type of Service/Traffic Class field), there is a new 20-bit Flow Label header. This header may be set to any random value by a sender, such that routers on the path could react to that value if its significance is signalled to them in advance. By the flow label definition, the 20-bit value should have no implicit meaning, and any nodes not supporting the flow label should simply set it to zero. The second aspect is how IPv6 QoS can be delivered. In principle, where IPv6 is deployed dual-stack alongside IPv4, implementations should treat the IP versions

equivalently, such that the same DSCP values are applied to the same preconfigured queuing behaviours. However, if IPv6 is tunnelled in IPv4 (e.g. by GRE tunnelling), any end-to-end QoS handling would require that the IPv6 DiffServ semantic is preserved in the outer IPv4 header, and handled accordingly by the intervening IPv4 network.

2.3 Alternatives to Classic QoS Methods

The main focus of this Technical Guide lies in the application of classic QoS methods by which IP packets are marked, classified and/or prioritised based on particular criteria that the packet may match. Such traditional techniques are described above (IntServ and DiffServ in particular). The aim of such prioritisation is to allow certain applications or traffic to get a higher quality of service than others, and in so doing gain a desirable quality of service for that application in the face of limited available bandwidth.

However, there are also other methods that can be used to conserve bandwidth, and thus reduce the likelihood of congestion or degraded service to certain applications. These may be used individually or in combination with traditional QoS methods. Examples of alternative bandwidth conservation techniques include:

- *Caching*. A cache device is placed between client devices and the services they are trying to access. The clients direct their requests via the cache (which may be done by manual or automatic means, often transparently to the user), and the cache in turn fetches the data from the true destination. The cache then keeps a copy of the data for a certain period in addition to relaying the data to the client. If the same data is requested again by any client, the data is served from the cache, and thus no external bandwidth is used. Caching is a popular technology, particularly for web content. However, with more and more content being served dynamically, the ‘hit rate’ on caches and thus their efficiency is generally falling. Any saving in bandwidth fees needs to be balanced against the cost of configuring and maintaining a robust caching system.
- *Traffic management (shaping/throttling).* The basic principle of traffic shaping or throttling is to limit the available bandwidth to a specific device or application. Its most common usage is probably in scenarios where rate limiting a user or device’s capacity on a service results in a fairer service for all, e.g. rate limiting data connections in a hall of residence network, or on systems connecting to a VPN service. In some cases, shaping may only be applied where a user or device has been deemed to have ‘misused’ bandwidth, perhaps exceeding a set monthly quota, and shaping is then imposed as a form of penalty or deterrent towards future behaviour.
- *Content filtering.* By filtering traffic based on content, certain traffic is able to use bandwidth that is denied to other traffic. Most sites will have some form of firewall device installed, with an associated policy. Filtering is simply a means to enforce that policy, e.g. to deny use of identifiable peer-to-peer applications like BitTorrent. In such cases the blocking or filtering may be assisted by use of an intrusion detection system (IDS), e.g. the Snort open source IDS can detect BitTorrent flows and disconnect devices by sending a message to the firewall to block traffic from the ‘offending’ host. Filters may be applied at certain times of day, typically being less restrictive outside of working hours.
- *Data compression.* While many applications have a good level of built-in data compression, there are many that do not. Data compression methods can be deployed on a bespoke basis, or may be used ‘on the fly’ within certain devices. There may also be opportunities to enable data compression in bandwidth hungry applications that have the option to use it. Compression ratios will depend on the data in question, but may

often be as high as 10:1. (However, the process of compressing and decompressing data can add to overall latency.)

- *Content delivery infrastructure.* A content delivery network (CDN) is a form of data caching. In general, data is pushed from a central store to distribution nodes on the network. The aim of the CDN is to shift data transparently around the infrastructure to minimise the bandwidth requirement in delivering the content (usually video based) to the end user. Data may typically be distributed around nodes at night, or other hours of lower data usage on the network. A CDN might typically be deployed by a large ISP (like JANET(UK)) rather than individual sites, but where deployed can make a significant difference. An alternative form of CDNs are peer-to-peer networks which use cooperative caching and delivery mechanisms to distribute content, as seen in the BitTorrent system.
- *IP multicast.* Multicast enables multiple nodes to receive the same data in such a way that only one instance of that data ever passes over any given link on the network. In contrast, unicast data delivery or streaming requires one copy of the data for each receiver. A special set of IP protocols and addresses are required to support multicast, and deploying and troubleshooting multicast can sometimes be complex. However its benefits, particularly for streaming live video/audio events, are significant, e.g. a site with a 10Mbit/s uplink can make a 1Mbit/s stream available to 100 receivers by only using 1Mbit/s of its uplink capacity. Without multicast, that is simply not possible.

These methods are particularly attractive where relatively low bandwidth links exist between sites and their upstream network providers. There are products available that often offer combinations of these technologies in single devices that can be deployed on a site border (e.g. caching, shaping, data compression and filtering).

2.4 Arguments For and Against Over Provisioning

There are several ways to avoid delays and loss within a packet-switched network, each with its own advantages and disadvantages. The most popular are network over-provisioning and traffic management/engineering.

2.4.1 Over-provisioning

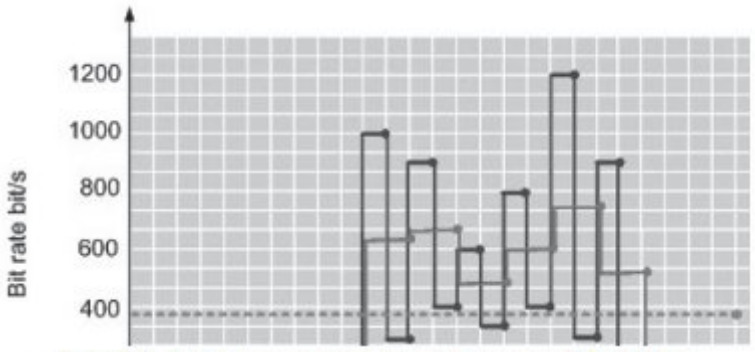
Over-provisioning means that there is sufficient bandwidth to satisfy all types of traffic at any time and any part of a network, such that it never becomes congested and therefore packets never experience delays in queues or loss because of queue overflow. (Of course, packets may be dropped by the network for reasons other than congestion; for example because of errors on a faulty link.) Simplicity is the key property of over-provisioning as it means that the existing IP transport is sufficient to handle all traffic without delay or loss. Moreover, router configuration complexity is kept to a minimum and operational stability is increased simply by the fact that the equipment is using one less feature. In an extreme case, a poorer level of QoS may be given to all traffic if the extra processing load on a router caused by the use of QoS is sufficient to cause further lost or delayed packets. It is worth noting at this point that this description of an over-provisioned network ignores the random aspect of network behaviour and is deliberately simplified to emphasize the effect. A more precise description of an over-provisioned network would be to say that events such as packet delays and loss due to queuing occur very rarely (i.e. their probability is very low) and hence all types of applications, including real-time, are supported by available network performance.

However, it may be difficult to prove that a network is really over-provisioned, and that all traffic will not experience some delay or loss. Only the constant monitoring of packet delivery times can definitively prove that traffic does not experience delay and loss and it is extremely difficult to take account of the large numbers of flows passing through large networks such as JANET. It is therefore more likely that some selective monitoring of ‘suspicious’ interfaces or traffic streams might be conducted where necessary.

As permanent monitoring of QoS metrics to gain explicit proof of over-provisioning is a challenge, another approach has proved very popular in practice: monitoring link utilisation. This is a much simpler technique to estimate whether or not a network or particular link is over-provisioned. In this approach, a link may be considered overprovisioned if its average utilisation level is permanently low, for example, lower than 10%. Such sustained low values will almost certainly guarantee that any short-lived bursts of traffic will not cause delay or loss to traffic. However, it is still possible that such a burst of traffic could take the link utilisation to 100%, and so for that instant packets may be delayed or lost.

Unfortunately, it is difficult to detect such incidents as routers will typically report real-time link load as an average over time; back-end statistics gathering and analysis tools generally also behave in the same way. As such, if occasional bursts of traffic exceeding line capacity occur in the order of milliseconds or less, they will never appear in statistics as they will be lost in the averaging process. Figure 2-6 illustrates this point in a crude fashion by charting fictional line utilisation against time for a 2Mbit/s link. This diagram shows three different curves, obtained on the basis of the same traffic but with different intervals of averaging of traffic data.

Figure 2-6 - Differing sampling rates impact of measuring link utilisation



[6]

A network monitoring system taking utilisation readings every 1ms is shown by the dark line. The grey line plots the results by averaging the readings over 2ms periods, and the dotted line shows the results of the average of all the readings over a 25ms period. Common practice for the above figure would suggest that utilisation above 512Kbit/s (about 25% of the link capacity) would be the point at which the link could no longer be considered over-provisioned The dark line shows that the link is definitely not overprovisioned as it is running well over 25% of its capacity during six intervals of the whole time period of observation. One can therefore expect that packets during these intervals may have experienced delays and possibly some have been dropped. Despite being based on the same readings, the grey line shows a noticeable difference (lower peak figures) and a network manager would be less concerned about congestion on the link based on these figures. Finally, based on the dotted line, the link clearly appears to be within the bounds of over-provisioning.

The figures used to create the graph are artificially extreme but do illustrate one of the more subtle issues that need to be considered. This is a common and very important effect of the influence of the averaging interval: a long averaging interval can result in the loss of a great deal of detail. In practice, measuring in terms of millisecond intervals is very hard, if not impossible, while in network terms a millisecond is a very long time. Obtaining meaningful statistics is therefore a key problem in attempting to understand and operate QoS on a statistically multiplexed network.

To evaluate over-provisioning, utilisation can be divided into three ranges:

- less than X%: a network is over-provisioned
- between X% and Y%: an ambiguous range; if possible, other metrics such as packet delay and loss values should be monitored, to check if delay or loss is occurring
- greater than Y%: a network is not sufficiently over-provisioned.

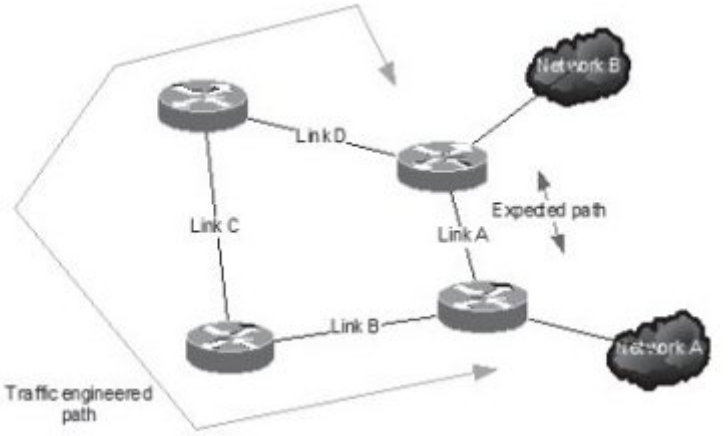
The general perception is that X% is somewhere between 10% and 25% and Y% is somewhere around 40% - 50%.

Over-provisioning also poses the problem of how to keep all of a network permanently overprovisioned. This obviously requires the continuous upgrade of link capacity which will usually also involve an increase in costs. However, experience and feedback (the JANET Network Performance Survey [JANETSurvey]) has shown that it is usually a site access link which suffers from congestion over other networks, either in the Regional Networks or the JANET backbone.

2.4.2 Traffic Management/Engineering

Traffic management may be a useful technique where there are clearly identifiable causes of congestion in a part of the network. For example, if it is clear that a single source of content is causing congestion, it may be possible to deploy a distributed content delivery system, to spread the traffic load more evenly over the network.

Traffic engineering is a form of traffic management where IP packets take a path through the network other than that suggested from the physical network topology. By manipulating routing protocol metrics, or using tunnelling mechanisms such as IP-in-IP, MPLS or the emerging PBB-TE, it is possible to direct traffic away from congested links over a different path. This is usually physically sub-optimal (for example, over a longer distance, or through more IP routers), but may be a less-congested path. Figure 2-7 illustrates the concept, albeit in an extreme and artificial situation.



ngested routes

[7]

In this topology, one would expect IP traffic between networks A and B to use the direct link between the two routers they are attached to. However, if link A were for some reason suffering congestion, traffic engineering could force traffic to take the alternate route, via links B, C and D. In essence, this is a trade-off between loss and delay, and operational complexity. Attempting to troubleshoot a performance problem if such re-routing of traffic occurs frequently is hard, and there are several applications in use within the JANET community that are particularly delay sensitive.

Data compression techniques might also be deployed either in the form of in-line dedicated devices or by making use of features on existing routers. However, this will add delay to traffic and may place substantial load on the router CPU. In practice this may not reduce traffic as much as expected as many large data flows are already compressed before transmission. Where a network is experiencing continual growth, traffic management can only really be seen as a temporary measure, as demand for content will grow and the links used for traffic engineering tunnels or compression will themselves becoming congested. In this case, the only alternatives remaining are to upgrade capacity or deploy QoS. Deploying QoS does not in itself prevent congestion; however it does permit resources to be guaranteed for certain traffic flows (with stateful technologies such as IntServ), or traffic to be prioritised (as stateless architectures, such as DiffServ). In a sense, QoS deployment is a form of traffic management,

and in some circumstances will not necessarily be the optimal long term answer for a network that regularly suffers excess traffic load.

2.4.3 Managed Bandwidth Services (MBS)

The term ‘managed bandwidth’ reflects the fact that it is not about standard IP or Ethernet connectivity as these networks provide their clients only with best effort transport, with no guarantees in bandwidth. Bandwidth within a plain IP or Ethernet network is distributed between flows in a random fashion, and to change this and provide selected end nodes with a dedicated bandwidth, some additional management efforts are needed. These efforts include mechanisms and techniques similar to those that are used for QoS, providing, for example, admission control, traffic classification and policing. This is quite understandable as MBS low packet latency and loss is about providing proper bandwidth shares to selected traffic flows or classes; in other words QoS involves bandwidth management. The major difference between QoS and managed bandwidth service is that the latter doesn’t directly deal with latency and loss guarantees; it is only about providing clients with some fixed bandwidth (which doesn’t exclude good QoS characteristics but it is not a mandatory feature).

MBS is about providing some particular user applications, nodes, subnets or sites with connections that have dedicated bandwidth. The service is targeted towards types of network application that work inadequately unless the network provides them with some guaranteed amount of bandwidth. While many types of applications can benefit from guaranteed bandwidth, there are some that on the one hand are very important for users and on the other hand really require some fixed amount of bandwidth. Examples of such applications can be found in research communities when research centres need to exchange large volumes of experimental data, sometimes in real-time. MBS can also be used to provision resources for new network services and protocols.

MBS can be provided not only in packet-switched but also in circuit-switched networks. SDH/xWDM/OTN-based bandwidth channels have dedicated bandwidth, as these are circuit-switched technologies for which this is an intrinsic feature. Another important feature of bandwidth channels based on circuit-switched technologies is that each circuit is reliably protected from impact on other circuits as a principle of its operation. In packetbased networks we can only speak about statistically guaranteed bandwidth, meaning that the bandwidth provided might deviate randomly from its average level. Such fluctuations of bandwidth may not be welcomed by users but are a consequence of the packet principle of data transmission. The probability of significant bandwidth deviations should be sufficiently low to make packet bandwidth channels acceptable for users.

2.5 What DiffServ QoS Will / Will Not Provide

Introducing QoS on a network can improve the effective use of network bandwidth by reallocating existing bandwidth between applications according to their requirements. No extra bandwidth is put onto the network and no attempt is made to decrease or artificially re-route traffic. In this case, QoS simply attempts to handle different types of traffic in different ways, for example by prioritising real-time traffic ahead of a large file transfer. As discussed above, over-provisioning simply adds more bandwidth to the free-for-all competition for resources between all types of network traffic.

There is a common misconception that QoS is equivalent to bandwidth, so the higher the

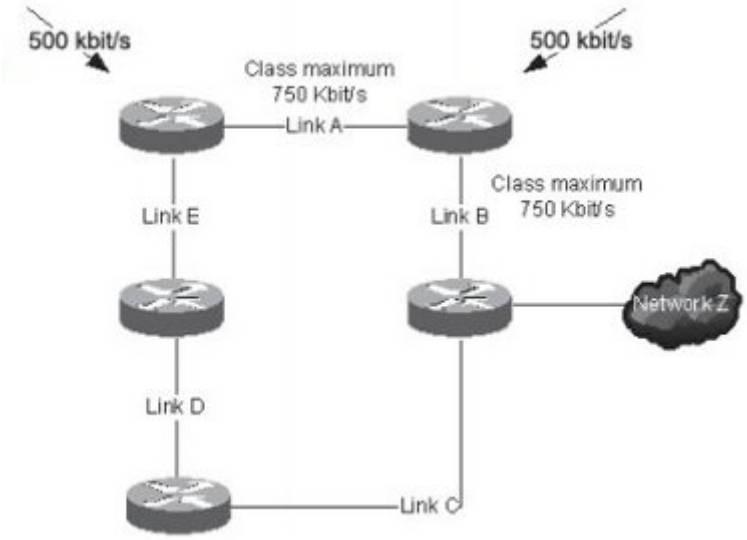
bandwidth available to a class of traffic, the better the QoS must be. While high bandwidth may indeed provide a good QoS by way of over-provisioning, if an application requires (and receives) 1Mbit/s of prioritised capacity from a network to meet its needs then the network has done its job and the fact that there may be 9.999Gbit/s of capacity left for other traffic to share does not mean that other traffic is receiving a better QoS. It simply means that under congestion, the application is more likely to receive the 1Mbit/s it requires. QoS can also play a useful role in fine-grained control over latency, even on a lightly loaded network. In this scenario it is not packet loss that is being protected against but a desire to route packets to their destination with the absolute least delay possible.

The deployment of QoS precludes the need to add bandwidth to a network, so it can also be seen as an economical way of handling the effects of congestion. However the costs of operating a network are not simply related to the costs of its links and configuring QoS adds complexity to a network, even more so in a multi-management domain network such as JANET. Additional complexity means more scope for problems to occur, whether down to faults/bugs in an IP router or operational errors. IP routers are complex devices in general and requiring one to perform yet more tasks raises the risk of interaction problems within its operating software. Network engineering staff will also require the skills to configure, maintain and troubleshoot QoS. QoS configuration and implementation differs from vendor to vendor, and in some cases the implementation and outcomes of the same configuration on different models of router will yield different results. The size of the average router configuration may also increase significantly with the addition of QoS. When DiffServ QoS was deployed on an experimental basis under SuperJANET4, the size of the router configuration files typically doubled in size, which introduced more scope for simple configuration errors.

QoS deployment as a service needs tight control over privileged bandwidth use as, if there is no control of ingress traffic, the network is open to abuse, even if it is not with malicious intent. If access to an elevated class of service were to be open, that effectively reduces that class to a best-effort service as anyone could transmit any amount of traffic into that class. Depending on the design, an open access elevated class of service may actually give worse QoS if the bandwidth allowance for a class fills up and excess packets are dropped or treated as besteffort.

In Figure 2-8, both links A and B have been configured to allow a maximum of 750Kbit/s of a certain class of traffic with the network permitting open access to this class. Two sources inject 500Kbit/s of traffic into this class, destined for network Z. Assuming no other traffic on the network is using the class, link A will carry its 500Kbit/s without problem; however once traffic from both sources gets to link B, packets from both sources will be dropped or delayed as the aggregate of the two streams is 1Mbit/s (greater than the configured bandwidth). If these links are otherwise underutilised 2Mbit/s links, it may be that this results in worse QoS than the default. In this case, it may be preferable for one of the sources to transmit traffic over link E-D-C if sufficient bandwidth is available (even if no traffic classes are supported).

Figure 2-8 - QoS resulting in network congestion and worse QoS



[8]

Open access to an elevated class of service is also clearly an open invitation for denial of service attacks and so a successful QoS design obviously requires very careful thought. This involves much more than simply enabling QoS with the default parameters on network equipment and there are many issues to consider:

- **Bandwidth assignment and access control** to each class of elevated service are critical elements of a network-wide QoS deployment. Bandwidth assignment is relatively straightforward to implement, a router can police traffic on ingress to ensure that it does not exceed the class bandwidth assignment, and routers are also capable of shaping (smoothing) traffic sent into a network. Typically these functions are done at administrative domain borders such as site access links, the links between Regional Networks and the JANET backbone, or a link from JANET to an external network.

The decision on how much bandwidth to assign to a class is more complex, particularly considering that this decision has to be enforced on every link in the network. This might involve determining the percentage of bandwidth assigned to a particular class on each link including aggregation concerns in the network backbone. For example, a central voice or videoconferencing hub is likely to require a higher percentage of bandwidth for elevated QoS on links where traffic from multiple sources starts to converge towards it. If this type of calculation is wrong, and traffic exceeds the assigned bandwidth, a poorer voice/video service will result because of lost or out of sequence packets.

Besides the limits of privileged bandwidth, access control is another important point which determines what flows are allowed to consume privileged bandwidth which is guarded by policing. Generally, the policed limit of bandwidth and selection of flows which are allowed to use this bandwidth are related issues. The crucial point here is to achieve a proper balance between these two sides such that the aggregated rate of all selected flows does not exceed the configured policing limit of bandwidth. The weak link of DiffServ as described in section 2.1.3 is in its inability to provide an end-to-end solution to keep this balance; traffic classes are uncertain in terms of aggregate rate by definition. IntServ architecture provides such a solution controlling every single flow but it is not scalable enough, especially for core networks.

- **Monitoring and management** are clearly also important as the growth in use of voice/video traffic may require an increase in the maximum bandwidth available to the

class. While increasing the bandwidth for this class is straightforward, the effect this has on other traffic on the network also needs to be considered. Keeping a good level QoS for the voice/video service has a noticeably negative impact on other users of the network so it may be sensible to devise a model to trigger link capacity upgrades once the maximum bandwidth assigned for a particular class of service approaches a certain threshold. The support of QoS-enabled networks thus involves an ongoing process of reacting to the changing needs of user applications, so that the QoS implementation continues to provide a consistent service.

- **Combining approaches.** It is also reasonable to combine these different approaches, for example DiffServ QoS and over-provisioning, particularly when dealing with large, multi-domain networks such as JANET. Where it is difficult to maintain overprovisioning, it is sensible to deploy QoS to work around the effects of congestion. This hybrid approach is the one that has been adopted in the current JANET QoS Model [JANETPolicy].

Links
[1] http://community.ja.net/system/files/images/tg-qos-2.1.jpg
[2] http://community.ja.net/system/files/images/tg-qos-2.2.jpg
[3] http://community.ja.net/system/files/images/tg-qos-2.3.jpg
[4] http://community.ja.net/system/files/images/tg-qos-2.4.jpg
[5] http://community.ja.net/system/files/images/tg-qos-2.5.jpg
[6] http://community.ja.net/system/files/images/tg-qos-2.6.jpg
[7] http://community.ja.net/system/files/images/tg-qos-2.7.jpg
[8] http://community.ja.net/system/files/images/tg-qos-2.8.jpg