Quality of service in BT’s MPLS-VPN platform

S F Carter

The multi-protocol label switching virtual private network (MPLS-VPN) is now firmly established as the preferred choice for providing private IP networking services for a very wide range of customer type, from small regional organisations, to large multinational corporations. Quality of service (QoS) is a natural complement to MPLS-VPN technology, enabling multiservice operation, and supporting the increasing drive toward application convergence. BT has recently launched a second-generation QoS scheme for its MPLS-VPN platform that gives unparalleled capability. This paper gives a detailed overview of the principles and mechanisms of differentiated-services QoS, and describes how this is applied in the BT design.

1. Introduction to MPLS VPNs and QoS

The last few years have seen spectacular growth in MPLS-VPN (multi-protocol label switching virtual private network) services. The major benefit of MPLS-VPN technology is that it allows large numbers of independent VPNs belonging to different customers, to be provisioned on a common core infrastructure, allowing major economies of scale. Each VPN has independent IP address-space and isolated routing, while a range of options for inter-site connectivity is supported to meet different requirements, including full-mesh ‘any-to-any’ connectivity. This is in contrast to traditional private IP networks that use dedicated routers interconnected over private circuits, so that core bandwidth and connectivity is usually heavily constrained by cost. MPLS-VPN technology therefore represents a major advance that enables highly cost-effective IP network solutions across a very wide range of customer types, ranging from small regional organisations with only a few sites, to large global multinationals with many hundreds of sites.

Following on from the rise in affordable private networking that MPLS-VPN technology brings, a key trend is now that of convergence — the desire to consolidate a range of applications, including voice, over a common IP service. Convergence means much more than simply replicating the services previously obtained from separate networks to achieve equivalent services from one network. The real benefit comes from the opportunities to integrate and enhance these applications, and to combine new ones such as multimedia.

A major challenge to convergence is that applications differ enormously both in their traffic characteristics, and in what they require from the network in terms of performance (delay, packet-loss, and jitter). Performance degrades rapidly following the onset of congestion, but this is an unavoidable fact of life in packet-based networks. Indeed, traditional transmission control protocol (TCP) based applications (e.g. file-transfer, e-mail, Web) are almost guaranteed to produce at least occasional network congestion, since they are designed to maximise throughput by adjusting their sending rates to the limits of network capacity. These same applications are also tolerant to the overall congestion produced by other traffic, the effect being to cause a graceful reduction of throughput, rather than an abrupt halt. But there are other classes of application where congestion is a much more serious problem. These include real-time services, such as voice over IP (VoIP) and multimedia, which have very stringent requirements and simply become unusable in the face of congestion. Also included in this category are time-sensitive interactive data applications, and certain data applications where it is essential that a minimum level of throughput must be maintained, rather than adapting in response to the growth of some possibly less mission-critical traffic source. Therefore, the challenge of convergence is to carry all the traffic, while simultaneously ensuring that the goals of the more performance-critical applications are met.

The problem can in principle be solved by over-provisioning — ensuring that the available bandwidth is sufficient to meet the needs of all applications, so that the most stringent applications receive the performance they need. But in the majority of cases, especially where VoIP is required, the degree of over-provisioning required would make this prohibitively expensive. Instead, a much more cost-effective solution is QoS which, in essence, means giving more favourable treatment to the most performance-critical fraction of the traffic, isolating it from the effects of overall congestion. QoS is not a new concept in IP networks, and a range of different architectures and detailed mechanisms have been proposed to support it. But none of these were at all widely adopted, until the work of the IETF Differentiated Services Group, who defined a framework for a relatively simple and scalable approach to QoS, which has become known as 'DiffServ'. This has been widely embraced by both equipment vendors and service providers, and the very large majority of QoS deployments in use today are based on the DiffServ model. DiffServ is fundamentally a 'subscription model' and perfectly complements what is required in a MPLS-VPN network, though its use is not at all confined to MPLS-VPNs.

BT launched its global MPLS-VPN service in 1999, based initially on a DiffServ design offering four classes of service. Since then, the size of this network has grown enormously, convergence has emerged as a major driver, understanding of QoS has matured throughout the entire industry, and vendor equipment capabilities have increased dramatically. Therefore BT has developed a new QoS service, which was launched in October 2004. This offers more traffic classes (six user traffic classes plus a seventh, mangagement class), presents a richer set of customer options, and aligns more closely with IETF standards. This paper describes the new QoS model, beginning with a detailed description of the DiffServ approach. Note that in BT's MPLS-VPN product portfolio, the term class of service (CoS) is generally used in preference to quality of service.

2. QoS using DiffServ

The overriding principle of DiffServ is the separation of traffic into different components, or classes, which are then treated differently by the network, the best treatment being reserved for the most performance-critical classes. Analogies may be made with first-class and second-class treatment in other contexts, such as rail travel, or indeed with the postal system [1]. Unlike these examples though, DiffServ is not limited to only two classes. The IETF architecture, recommendations for deployment, and other supporting documents are given in a series of RFCs [2—4]. The following is a concise summary of DiffServ, with some paraphrasing of IETF terminology.

Figure 1 illustrates a DiffServ network. This example here shows a number of customer sites interconnected over a common network, where each site has

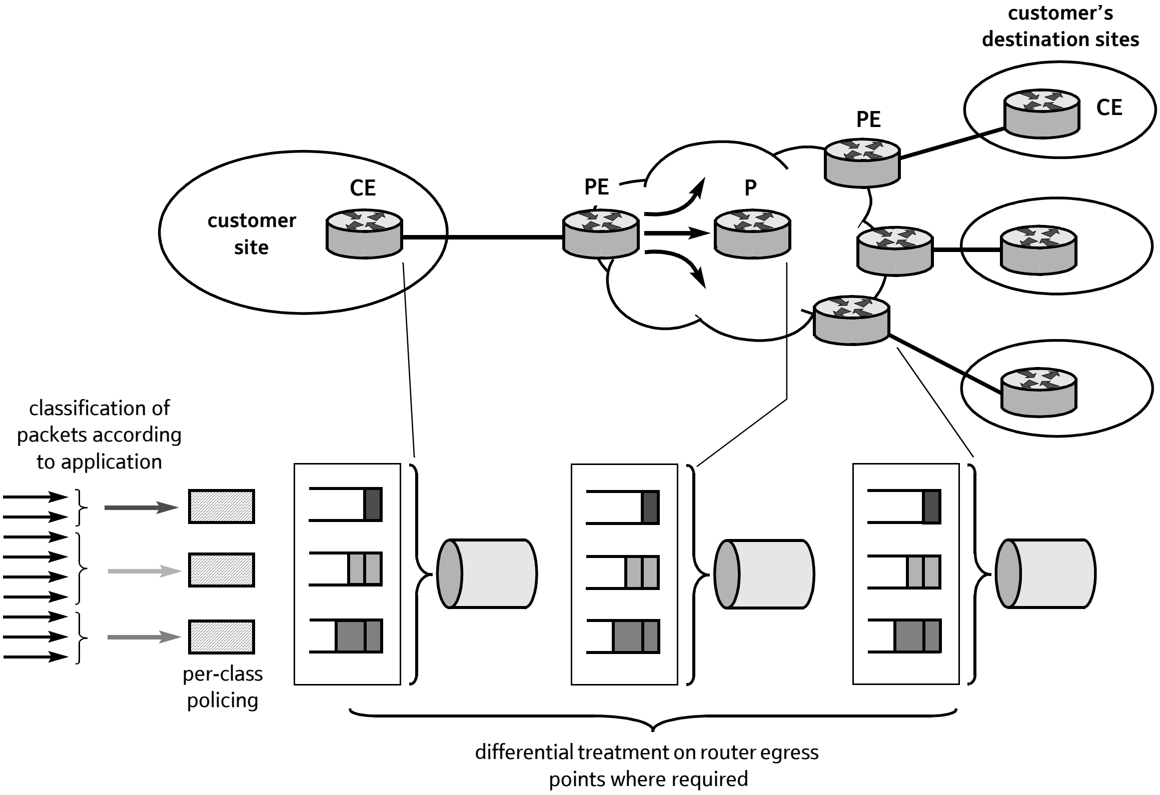
customer's destination sites
CE
PE
CE
PE
P
customer site
classification of packets according to application
per-class policing
differential treatment on router egress points where required

Fig 1 Showing DiffServ architecture in an MPLS-VPN context.

connectivity with every other. This corresponds to a typical customer MPLS-VPN. In the MPLS-VPN case, the router within the customer premises is known as the CE (customer edge) router, and each of these is connected to an adjacent router in the service provider domain, known as the PE (provider edge) router. Other routers internal to the service provider network are known as P (provider) routers. Though this figure illustrates specifically an MPLS-VPN context, the DiffServ definition is broad, and the principle is applicable to other types of network as well.

Figure 1 shows the three key functional components of the DiffServ solution — classification, policing, and differential treatment — which are described separately below. If correctly applied, and given sensible control of traffic levels, then these three components enable control of the end-to-end performance within a multi-service network.

2.1 Classification

Classification is the process by which individual packets are assigned to each service class as they enter the network. It relies on the existence of a set of rules by which packets originated by different applications may be recognised. The set of classification rules used at each site is chosen uniquely to meet the requirements of each customer, according to the particular applications in use, and the relative importance placed on them.

Classification involves looking inside the packet at both network headers (IP header) and transport layer headers (TCP or UDP headers) to identify particular values of certain fields contained in these from which the application may be deduced. These fields might include IP source or destination address, protocol ID, TCP or UDP port numbers, and values of the particular field associated with QoS, known as the DiffServ codepoint (DSCP) field (see section 3.1). In many cases, applications may be recognised by particular static values of these fields that occur in every packet for that particular application. For example, applications such as VoIP are usually able to mark the DSCP value, others may use particular port numbers, while applications running on a dedicated server may often be identified through source or destination IP address. In some cases though, it is not possible to recognise particular applications from static field values, and so-called stateful inspection is necessary instead. Typically this is because applications do not use predetermined port numbers, but negotiate new values every time a session is initiated. Stateful classification involves listening in to the negotiation phase, and identifying the port number every time a new flow is initiated.

Classification can be an intensive process, because it involves detailed inspection of every packet. Therefore it is done only once, at the edge of the network, usually on the CE router. Once the class has been determined, packets at the CE are marked with class information, by overwriting the DSCP field in the IP header. Subsequent network elements can then identify the designated class simply by looking at the DSCP. In practice, this works slightly differently in an MPLS VPN, but the principle is the same — in an MPLS-VPN, packets are carried across the core encapsulated within a stack of two MPLS labels (the DSCP field within the IP packet itself is not visible to core routers, so the equivalent field within the MPLS label, called the exp field, is used instead). Encapsulation into MPLS takes place at the PE router, and at this time, the value of the exp header may be set with a value derived from a simple mapping of the DSCP.

2.2 Policing

A vital part of engineering any network for performance is control over the volume of traffic admitted at the network edges, in order to limit the potential for congestion. Policing is the mechanism used to provide this control. Policing is particularly necessary for the 'premium' classes, since their delay and packet-loss performance depends on limiting their respective traffic volumes to defined fractions of the available bandwidth on every link in the end-to-end path. Therefore, in a DiffServ network, at each traffic-entry point, there is a police element for each service-class to regulate the volume of traffic allowed in. This aligns well with the subscription model, whereby 'premium' classes have a higher per-kbit/s charge — customers specify how much bandwidth they require at each site for each class, and their traffic is policed to prevent these levels being exceeded.

In the BT MPLS-VPN service, the highest performance class is the 'voice' class, and is therefore the most expensive. The action of the voice police element is to drop packets that exceed the specified 'in-contract' bandwidth. In practice, this action should seldom occur, since it is expected that the customer is able to specify exactly the bandwidth required — each voice call generates a well-defined traffic rate, while voice equipment can be configured to control the maximum number of voice calls initiated or terminated on each site.

Voice is not the only 'premium' class service. Performance-critical data applications, or ones simply requiring strong bandwidth assurances must also be considered, and also require 'special' treatment. With many data applications it is usually not possible either to characterise or control aggregate traffic rates with anything like the rigorous precision possible with voice — simply dropping any 'out-of-contract' traffic would tend to lead to high levels of dropped packets. Instead, in the BT design, the police element applies 're-colouring' to out-of-contract traffic. This involves labelling the in-contract and out-of-contract packets differently, so their status is visible to subsequent routers in the end-to-end path. If congestion levels within the premium class are not too high, then both types of packet are forwarded; but should it become necessary to drop any packets, then 'out-of-contract' traffic is dropped in preference to in-contract traffic. The mechanism to achieve this at subsequent congestion points is weighted random early discard (WRED), which is discussed in the next section.

2.3 Differential treatment — scheduling and intelligent discard

As has been described, the first two DiffServ components are concerned with identifying, labelling and regulating volumes for the traffic within each class. The final component is concerned with providing appropriate differential treatments between classes. This is necessary at every point in the network where there is the potential for congestion to occur, and these points are usually on the output interfaces of the various routers within the core and access parts of the network. Large packet-queues and packet-loss are the direct result of congestion, but it is the aim of QoS to eliminate or reduce this for classes containing the most performance-critical traffic. Two distinct mechanisms are commonly used in combination to achieve this:

- differential queuing, also known as scheduling,
- intelligent discard mechanisms within some queues.

Their joint use is illustrated in Fig 2.

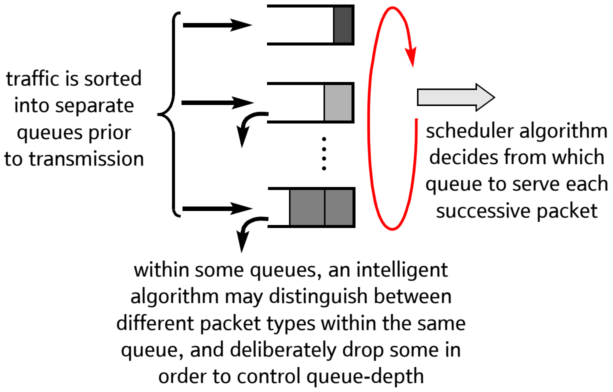
traffic is sorted into separate queues prior to transmission
scheduler algorithm decides from which queue to serve each successive packet
within some queues, an intelligent algorithm may distinguish between different packet types within the same queue, and deliberately drop some in order to control queue-depth

Fig 2 Applying differential treatment at router egress points.

2.3.1 Scheduling

Differential queuing involves giving each class its own dedicated packet-queue, instead of a single shared queue. Packets are then served from the set of queues under the control of a scheduling algorithm. The operation of the scheduler is as follows:

- every time a packet has been transmitted, a decision must be made from which queue the next packet should be sent,
- by configuring the scheduler to favour some queues over others the bandwidth available to each class can be controlled.

Broadly, performance is controlled by managing the ratio of traffic-demand to available bandwidth for each queue. It is a key part of the job of designing and managing a DiffServ network to specify the bandwidth of each queue in order to achieve the desired behaviour.

The literature contains much material on the design and relative merits of different scheduling algorithms, and there is sometimes a trade-off between the precision of control possible versus complexity of the implementation, a factor that is particularly relevant as the industry trend is toward routers with ever-higher port densities and throughputs. A careful assessment of characteristics of the scheduler used in any proposed router is important to ensure it meets the requirements of the QoS service.

It is usually considered an essential requirement that a scheduler should be 'work-conserving', which means that if any particular class is not using its entitlement, then other classes may share its allocation. This supports a model where, to meet performance goals, critical classes are given a generous allocation of bandwidth, which may often not all be used by this particular class. If the scheduler is work-conserving, it means that this spare bandwidth is available to be used by other classes. This is known as bandwidth-borrowing.

2.3.2 Intelligent discard (e.g. WRED)

In the absence of any specific discard mechanism, when a queue is congested (i.e. packet-arrival rate exceeds the rate at which packets may be transmitted), the queue grows to the limits defined by the available buffer-space, while beyond this point, excess packets are dropped. Such 'tail-drop' behaviour can be undesirable for the two reasons given below, and instead intelligent mechanisms may be used to provide more optimum behaviour:

- TCP traffic (i.e. the large majority of data traffic) does not always respond well to the abrupt onset of packet-dropping that occurs with tail-drop behaviour,
- when carrying both in-contract and out-of-contract traffic within the same queue, it is necessary to ensure that, if any packets must be dropped, out-of-contract packets should be dropped in preference to in-contract packets — this

complements the out-of-contract policing action described above.

Within the industry, by far the most widely used intelligent-discard mechanism is WRED, which involves specifying a 'profile' that defines the relationship between drop rate and queue depth [5]. Rather than the onset of dropping occurring in an abrupt fashion, it can be configured to begin gently, increasing only as queue-depth increases further. This behaviour is known to complement the congestion-avoidance mechanisms built into TCP and can lead to improved efficiency. Different WRED profiles can be configured for different components of traffic within the same queue, such as the in-contract and out-of-contract components, which may be identified though the different DSCP markings.

It might be though that an alternative to carrying in-contract and out-of-contract traffic within the same queue and discriminating with WRED, would instead be to place these in different queues, perhaps with the out-of-contract component in a 'best-effort' queue. The reason that this is not done is that it would tend to lead to relative re-ordering of in-contract and out-of-contract packets, which is highly undesirable for packets within the same application flow. Any degree of re-ordering creates more work for the transport protocol, while excessive re-ordering can lead to time outs and other problems.

3. IETF DiffServ definitions

3.1 DiffServ codepoint

Apart from promoting the overall architecture described above, the IETF DiffServ working group made two other key contributions. The first was the definition of a six-bit field within the IP header, called the DiffServ codepoint, which is actually a re-definition of part of the original 8-bit type of service (TOS) field. Although a number of uses of the TOS field had been defined prior to DiffServ, only the leading 3 bits (known as 'precedence') had found any significant usage. The DSCP definition allows 6 bits (64 values) to be used to label class information, and gives backwards compatibility with precedence (see Fig 3).

3.2 DiffServ per-hop behaviours

The second major contribution of the IETF was to define some basic definitions that can effectively be used to describe different classes, although they do not use the term 'class'. Instead they refer to 'per-hop behaviours' (PHBs), meaning the per-class differential behaviour on a single router. But if PHBs are implemented in a consistent way in all the routers inside the network, then they result effectively in end-to-end service classes. The IETF has also recommended specific values of the DSCP field that should be used to indicate particular PHBs.

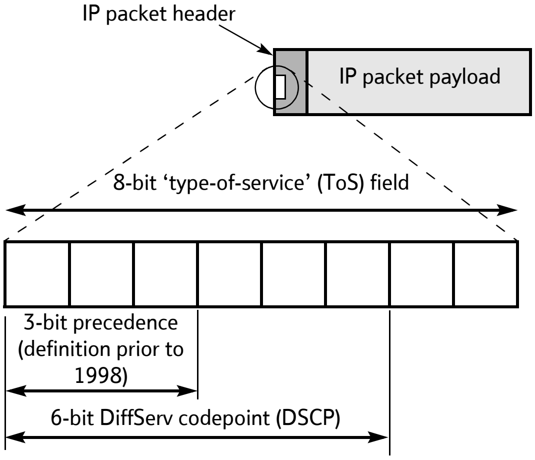
IP packet header
IP packet payload
8-bit 'type-of-service' (ToS) field
3-bit precedence (definition prior to 1998)
6-bit DiffServ codepoint (DSCP)

Fig 3 Definition of the DiffServ codepoint.

The most important PHBs that have been defined are expedited forwarding and assured forwarding.

3.2.1 Expedited forwarding (EF)

This is the PHB to be used for carrying traffic with the most stringent performance requirements, and is generally the one assigned to carry VoIP packets [3]. It is common practice for EF to be implemented using a priority queue, which gives the best possible performance. Priority queuing involves a very simple scheduler algorithm — after serving a packet, the scheduler looks to see if there are any packets waiting in the priority queue, and if so, the next packet is taken from this.

Providing the EF traffic rate is controlled (i.e. limited to substantially less than the link-rate), tight bounds may be met for packet queuing, and the corresponding delay, jitter, and packet loss. For more information on controlling end-to-end performance for VoIP, see the Appendix.

The recommended DSCP value for labelling EF packets is 101110. In practice, DSCP values are usually specified either by name (lower-case) or by the decimal value of the six-bit field. Therefore, the recommended DSCP value for EF is known either as 'dscp46', or simply as 'ef'.

3.2.2 Assured forwarding (AF)

The IETF has defined four groups of assured forwarding PHBs [4], known as AF4, AF3, AF2, and AF1 (the IETF actually uses the term behaviour aggregate, or BA, rather than group). Within each of the four BAs, three PHBs are defined, which have different levels of drop priority. This makes twelve PHBs in total, and each has a recommended codepoint.

Separate queues for each AF BA are mandated, with intelligent discard within each queue to provide the three levels of drop priority. This aligns with the description of combined scheduling and intelligent discard described earlier. When a particular AF queue contains only a few packets, then the expected behaviour is that these will remain in the queue until they are served. However, if sustained congestion occurs resulting in the queue filling beyond a certain point, then packets with the highest 'drop eligibility' will be dropped first. If the queue fills still further, then packets with the next level of drop eligibility will be dropped. Figure 4 illustrates this broad behaviour, while Table 1 gives the IETF recommended DSCP values.

The specific intelligent-discard mechanism to achieve this behaviour is not mandated by the IETF, but most router vendors have implemented WRED. An implementation is not required to support all four AF BAs to be considered 'DiffServ compliant', and there is no prescribed service difference between the four AF classes, i.e. AF4 does not necessarily deliver 'better' performance than any of the other AF BAs.

3.3 Absolute performance of EF and AF PHBs

The IETF DiffServ recommendations define an overall framework, terminology, and some basic building blocks. Detail for how these should be applied is left very deliberately to equipment vendors and service providers. Therefore, PHB definitions do not seek to mandate detailed scheduler behaviour, nor do they define specific numerical performance characteristics of PHBs. In any case, it is important to note that performance is not determined solely by scheduling or other treatment at the router, but is influenced in equal measure by the volumes and characteristics of the traffic the network operator or service provider chooses to admit to the network. Furthermore, each PHB or service-class cannot be considered in total isolation — at each network link there is only a finite amount of resource (principally bandwidth, priority for timely access to bandwidth, and packet buffer-space) that must be shared and apportioned. Therefore, dedicating more resources to one class reduces the resources available for the others.

Likewise, no requirements are made on how many different PHBs should be supported in a DiffServ domain, as this is likely to depend on perceived requirements. This deliberately gives substantial freedom both to equipment vendors and to service providers. In summary, for a DiffServ network, there is no industry-standard definition for either the number of different classes, or detailed end-to-end behaviour of particular classes. Substantial differences in approach can therefore be found between different vendor implementations, and between different service provider offerings.

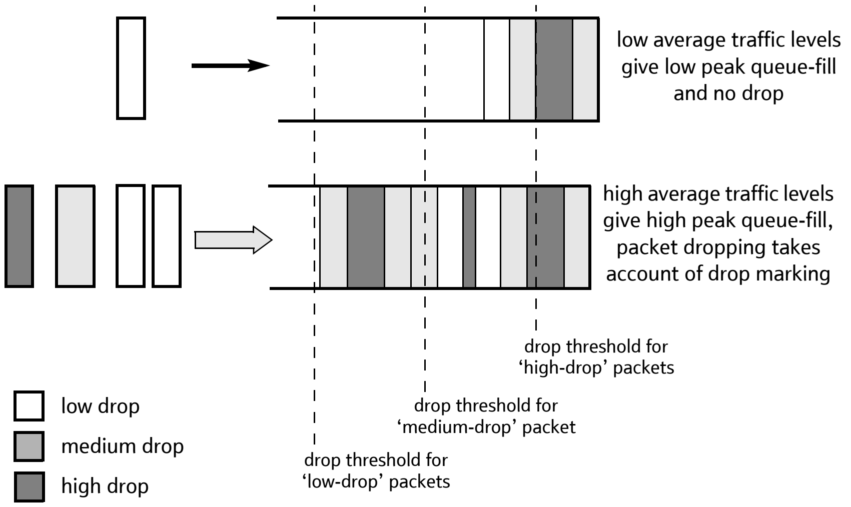
low average traffic levels give low peak queue-fill and no drop
high average traffic levels give high peak queue-fill, packet dropping takes account of drop marking
drop threshold for 'high-drop' packets
drop threshold for 'medium-drop' packet
drop threshold for 'low-drop' packets
low drop
medium drop
high drop

Fig 4 Treatment of traffic with different drop eligibility markings in AF queue.

Table 1 IETF recommended DSCPs for AF PHB set.

	AF4		AF3		AF2		AF1	
	Name	Decimal	Name	Decimal	Name	Decimal	Name	Decimal
Low drop	af41	34	af31	26	af21	18	af11	10
Med drop	af42	36	af32	28	af22	20	af12	12
High drop	af43	38	af33	30	af23	22	af13	14

4. How many classes are required?

This is not a question to which there is one simple answer. It depends upon how many performance-sensitive applications are present, and on their precise and individual requirements. It also depends on access speed, since the delay that accompanies packet congestion becomes much more of a critical factor as access rate is reduced. It is almost always the case that VoIP media packets (i.e. the packets carrying encoded voice) should be given a dedicated class — so the question is then: 'How many other 'premium' classes are required?'

A 'three-class' model is often adopted, consisting of 'voice', 'premium', and 'standard' classes, reducing to only two classes if voice is not required. This model has the advantage of relative simplicity, often fully meets requirements, and is certainly a large advance on a single-class model. Here, all non-voice performance critical applications share the same class, and this is a good approach if the aggregate traffic levels of these applications are such that sustained congestion within this class is very unlikely to occur. Under these conditions, all applications behave as if on a 'lightly loaded' network.

This model is also suitable if the 'premium' applications share similar performance requirements and degrees of 'mission criticality'. Other applications, which may well be equally mission critical, but are less performance sensitive, such as e-mail, WWW, and bulk file-transfer applications, may be placed in the 'standard' class.

The 'three-class' approach will not meet all cases though, especially in environments where there are significant numbers of different performance-critical applications with different characteristics. More classes are then necessary to provide control over the bandwidth available to particular applications, and to provide isolation of the behaviour of some over others.

Where many applications are present, it is unlikely to be possible to give each a dedicated class. There are both technological reasons (router hardware capabilities, limit on the number of class-labels available) and operational reasons (management complexity, and difficulty in characterising requirements of all applications accurately) for not doing this. Instead, the approach should be to group applications in a sensible way, and place these groups in a small number of different classes. The strategy for grouping different applications should be based on avoiding placing applications with incompatible requirements in the same class, following the four broad considerations below, which are given as general guidelines.

- Degree of importance

  One aspect of QoS is certainly to support applications that are sensitive to network performance, and this has been discussed extensively in this paper. But in some cases an application may be considered worthy of protection, not because it has any special performance needs, but simply because it is of paramount importance to the business needs of the organisation. Placing the most 'mission-critical' applications in a separate class or classes is a powerful means to ensure these applications receive the bandwidth they need at the expense of less-important applications.

- Bandwidth requirements — minimum requirements, and potential consumption

  Applications differ hugely in their requirements for bandwidth. Some applications are quite constrained in their behaviour (though may stop working if not provided with a relatively modest minimum requirement) while others (e.g. multimedia) may be capable of consuming larger amounts of bandwidth, and in some cases, they may be able to consume whatever bandwidth is granted if conditions are not carefully controlled. An example of a bandwidth-hungry application is video/multimedia traffic, and this behaviour may be exacerbated by either accidental misconfiguration, deliberate attempts by users to obtain better quality at the expense of higher transmission rates, or greater-than-planned numbers of simultaneous sessions. In such cases it may be impossible to maintain the required performance of the particular video application simply due to insufficient bandwidth, but placing such traffic in a dedicated class may at least isolate the problem, and prevent the performance of other applications from being degraded. Therefore, care should be taken to isolate particularly bandwidth-hungry applications from others.

- Sensitivity to latency

  Tight requirements for round-trip delay may be a requirement for many transactional applications. Such applications are not generally compatible with other applications that are likely to generate congestion, since congestion and queuing-delay are inextricably linked. Therefore, applications with a tight delay bound should be placed in a dedicated queue, and, to minimise delay, assigned a bandwidth that is greater than the likely peak demand. Unused bandwidth from this class will be made available to other classes via the scheduler's 'borrowing' mechanism. A well-designed application should not make arbitrary demands of the network; rather its demands should be closely coupled to the fundamental nature of the application. Therefore, the applications with the tightest constraints on delay are likely to be real-time interactive applications, such as VoIP, multimedia, or transactional data applications. A tight time constraint also implies that retransmission in the event of packet loss is not appropriate, and therefore these applications also often have fairly tight packet-loss constraints (athough, in the case of VoIP, embedded loss-concealment algorithms give some tolerance).

- Adaptive or non-adaptive nature

  The TCP transport protocol is inherently bandwidth adaptive, i.e. the application end-points are able to sense network congestion, and adapt their sending rates in response. As a result, in some cases, TCP applications are tolerant to congestion — slowing down in the face of congestion rather than simply ceasing to work at all, although in other cases, attaining a certain minimum bandwidth to make the application usable. In contrast, applications based on the UDP transport protocol often have no ability to adapt. It is generally not good practice to place adaptive and non-adaptive applications in the same class, because if congestion occurs, the non-adaptive applications are likely to gain access to almost all the bandwidth.

A first step to identifying how many classes are required, and to making a sensible grouping of applications, is to characterise the traffic on the network. A rigorous approach will involve a measurement process to capture typical usage profiles for all the applications running. The results of this process combined with an understanding of any special performance requirements for individual applications, enables a successful QoS strategy to be defined [6].

5. QoS in BT's MPLS-VPN product set

This section describes particular features of BT's MPLS-VPN second-generation QoS scheme. This follows the IETF DiffServ architecture, as described earlier in this paper. It supports up to six traffic classes, named EF, AF4, AF3, AF2, AF1, and DE (default), together with a seventh class, Management, specifically for carrying control and management traffic for CPE. By no means is every customer likely to require this maximum number of classes, and it is possible to order only a subset.

The four AF classes each support in-contract and out-of-contract traffic. To maximise flexibility, there are no predefined differences between these four classes in terms of bandwidth or performance, and no specific constraints or recommendations are made for which type of application should be placed in each. This is left entirely under the control of the customer, who is able to specify the necessary parameters to define their characteristics. Specific QoS features are described in the following sections.

5.1 CPE management options

For customers who order a VPN network service, two main options are supported. The 'unmanaged' or 'unbundled' option is for customers who wish both to provide and to manage their own CE router. As described earlier the CE device needs to perform both classification of each packet according to application, and marking the DSCP value for each class according to BT's specific marking scheme. Alternatively, the customer may opt for a solution where BT provides and manages the CPE as part of the overall service. Here, as part of the order, the customer must specify the set of rules that should be used by the classification process to map traffic from individual applications appropriately into classes. A third option is where BT provides an overall solution, rather than just the VPN network component. In this case any detailed specification of VPN QoS parameters will be managed as part of this solution.

5.2 Specification of QoS bandwidth parameters

The VPN supports a subscription model, where bandwidth for the EF, AF, and DE classes is priced differently. Customers must specify which particular classes they require and associated bandwidths, for each individual site. A high degree of customisation is supported for specifying bandwidth parameters for customers who require tight control. But with greater control comes greater complexity (the need for the customer to specify more parameters). Therefore, several ways of specifying bandwidth parameters are supported.

For each AF class, there are two parameters that specify the bandwidth configuration for the service:

- scheduling bandwidth,
- in-contract bandwidth.

Scheduling bandwidth controls the bandwidth share each class-queue obtains on the access link. Since the AF (and DE) classes also share access to any 'spare' bandwidth from under-used classes, the scheduler bandwidth is not necessarily the actual bandwidth a class gets, but is rather the minimum guaranteed bandwidth.

For the AF classes, the specified in-contract bandwidth is used to set the policer, and controls the maximum rate of 'in-contract' traffic that may be sent, with excess traffic being marked as 'out-of-contract'. As described earlier, 'in-contract' packets are better protected against discard in the core (via the WRED mechanism), and therefore (unlike scheduling bandwidth) a charge is made for in-contract bandwidth. For the EF class, because the out-of-contract action is packet-discard, the in-contract bandwidth and scheduling bandwidth are implicitly the same.

In general, the appropriate values for scheduling and in-contract bandwidth for a particular AF class are not necessarily equal. If it is considered essential that all the traffic is carried across the core with very low packet loss, then in-contract bandwidth should be at least equal to the scheduling bandwidth (or possibly greater, if significant 'borrowing' is expected). On the other hand, since in-contract bandwidth is charged, it may be considered acceptable to subscribe to a smaller amount of in-contract bandwidth, and rely on the fact that, in practice, the loss rate for out-of-contract traffic is normally small.

Customers may opt to specify both in-contract and scheduling bandwidth for the AF classes independently. Alternatively, they may prefer a simpler approach where they specify only the in-contract bandwidths, leaving scheduling bandwidth to be determined for them, based on a predefined formula. This second approach is likely to be completely satisfactory in many cases, especially where only one or two AF classes are required.

5.3 *Range of access types and speeds*

The VPN platform is a global one, and connection is possible through a range of access types. These include leased-line, Ethernet, ATM, Frame Relay, and xDSL (though not all of these are necessarily available in all countries). All of these offer (or are planned to offer soon) a QoS service using the same model. For each access type, a comprehensive range of access speeds is supported. For example, on leased-line, this extends from 64 kbit/s to STM-4 (622 Mbit/s). Access via dial-up connections is also supported.

5.4 *Fragmentation*

If class EF (i.e. for VoIP) is required at a particular site where the access connection is less than a certain speed, then fragmentation is employed to reduce the delay and jitter experienced by voice packets. Fragmentation is described in more detail in the Appendix.

5.5 *Transparency*

Although the DiffServ architecture and DSCP codepoint definition has been in place since early 1998, it is possible that there may still be some legacy applications that either use parts of the TOS field, or are sensitive to any network-induced changes to its value. This is believed to be very rare, and certainly contravenes IETF recommendations, but may still be an issue for some customers. To cater for such customers, the option of transparent operation is supported within the DE class. This means that DE class can accommodate a range of DSCP values (i.e. the values remaining that are not synonymous with any of the AF or EF classes a particular customer has ordered). Such traffic is carried transparently across the network, from end to end.

Alternatively, a customer may optionally select 'bleaching' to be applied to class DE, which deterministically marks all DE traffic to the same DSCP value of zero. This eliminates the possibility of unexpected re-classification of packets at a destination PE-CE link, caused by the source application marking the DSCP to particular values.

5.6 *Egress remarking*

If for some reason a customer has a DSCP marking scheme that does not align with the BT/IETF scheme, then the option exists to remark traffic as it leaves the CE and re-enters the customer domain. For this option, the customer should specify a one-to-one mapping of class to the required DSCP marking.

5.7 *Automatic classification of H.323 signalling*

In supporting VoIP, consideration needs to be given not only to the actual voice media packets, but also to the various signalling and control packets, essential for the set-up, tear-down, and maintenance of individual calls. BT recommends this traffic be carried in one of the AF classes (which particular AF class is a matter for customer preference). To facilitate classification of H.323 signalling traffic, the option is provided for the insertion of automatic classification rules for H.323 traffic. The customer need only specify the particular AF class within which this traffic should be placed. Since such classification takes place on the CE router, this option is available only for the 'managed' service.

5.8 *Service-level assurances (SLAs) and core reports*

QoS is largely about managing end-to-end performance. A crucial part of this is the performance across the core. BT's MPLS service offers assurances for the performance of individual classes across the core, and provides customers with published targets and actual performance reports.

5.9 *Site reports*

Conditions within a customer's network are very rarely static; as organisations evolve, the application mix changes, and traffic levels grow. To help a customer track network performance and identify the possible need for changes in either aggregate bandwidth or per-class bandwidth allocations, BT provides a set of on-line reports. Utilisation reports give traffic levels for each class for each site, while site-to-site reports give performance figures (delay, packet loss, and jitter) between selected sites.

5.10 Managed VoIP services

Customers may elect to manage their own VoIP network, in which case, as far as the VPN is concerned, VoIP is just another application. Alternatively they can choose to buy a managed VoIP service, accessed via their VPN. Here, BT manages the service, and uses infrastructure such as VoIP call-servers and PSTN breakout gateways located within the BT domain. This is a flexible approach, and fully supports customers making the transition from traditional PBX-based telephony to integrated IP technology.

6. Managing and operating a DiffServ network

6.1 Managing the performance of the core

This paper has so far focused on the network-layer mechanisms employed in a DiffServ network. Equally vital to achieving end-to-end performance goals is the specification and maintenance of tight bounds on the traffic levels within each class, for every individual link inside the core. Note that the core network provides any-to-any connectivity, and policing is applied only on volumes of traffic entering the network at any point, irrespective of where it exits. This poses a challenge for managing performance, since there is no explicit constraint on how customer traffic is distributed across individual links of the core.

Within BT's MPLS platform, the per-class traffic levels on every link are monitored continuously, and trend-analysis is applied, so that link bandwidths can be upgraded to pre-empt growth demands. In addition, topology-routing models are employed to pre-plan capacity requirements in some regions of the network. Using these methods, traffic levels are maintained below the levels needed to meet given performance criterion, including satisfying the published SLAs, and this performance is verified through monitoring using network probes, from which performance reports (delay, loss, and jitter) are derived.

6.2 Router performance considerations

A successful QoS design requires a good understanding of the capabilities of the routers employed in the end-to-end path. A particular issue is packet-per-second (PPS) performance, and VoIP requires particular consideration since it is usually composed of small-size packets (typically in the range 60 byte — 200 byte, depending on codec type), and so produces a significantly higher PPS rate for a given bandwidth compared with data traffic. Therefore, good characterisation of any PPS limits inherent in the routers employed becomes particularly important if a significant fraction of the traffic is VoIP. In designing BT's QoS service for the MPLS-VPN, extensive performance measurements were carried out to determine these limits. Such testing is the only reliable way to generate rules for selecting the most cost-effective CE router type for a particular situation, and also for determining rules for how much traffic may be terminated safely on each component of the core network (PE routers and core routers).

7. Conclusions

MPLS-VPN technology is perhaps the biggest success story in data networks in the last five years, and BT is firmly established as a leading service provider. Sustained growth and commensurate additions to the infrastructure have allowed BT's network to evolve to a high-bandwidth backbone network with global reach, and offering customers a rich choice of connection options for access. This has now been enhanced by the deployment of a state-of-the-art QoS model, providing BT's customers with a truly versatile multiservice platform, to meet the needs of today and of the future.

Appendix

Understanding and controlling end-to-end delay for VoIP

VoIP is the most performance-critical application in common use. ITU G.114 specifies that one-way ear-to-ear delay should ideally not exceed a value in the region 150 ms — 200 ms (depending on codec type) if perceived quality is not to be reduced. The delay budget for the network is usually substantially less than this figure though, since there are several delay contributions from components of the VoIP equipment itself, including algorithmic, processing, and framing delays at the transmit end, and jitter-buffer delay at the receive end.

Network delay may be sub-divided into transport delay, and router queuing-delay. Although every transport link and router contributes to some extent, often particular sections of the end-to-end path dominate. A common scenario is where the VPN core links are of relatively high bandwidth (say STM-1 or greater) while the access-tails are of relatively low bandwidth (say 2 Mbit/s or less). In this case, assuming a properly managed DiffServ design, the core delay is likely to be dominated by transport delays, while each of the two access tails is dominated by router queuing-delay. These two delay contributions are discussed in more detail below.

A1 Transport delay

This is the delay due to the underlying transport network. This is most significant in long-haul sections of the core network, particular between countries. Almost universally the physical transport medium for the core network is optical fibre, which gives a figure of roughly 5 µs/km, or 5 ms for every 1000 km. Often though, the propagation delay is significantly longer than would be expected simply from the linear distance between two nodes, because the underlying infrastructure may not closely match the logical connectivity, e.g. topologies such as rings may be employed, leading to longer distances. Transport delay can also be introduced by non-optical components of the transport network, e.g. ATM switches.

A2 Router queuing delay

Assuming voice is placed in a class that uses Priority Queuing (as in BT's EF class), queuing-delay consists of two main components, data-induced delay, and voice contention delay (see Fig A1).

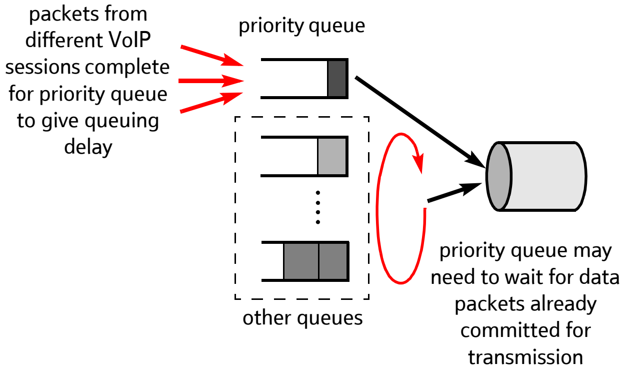
packets from different VoIP sessions complete for priority queue to give queuing delay
priority queue
other queues
priority queue may need to wait for data packets already committed for transmission

Fig A1 The two main components of router queuing delay.

A2.1 Data-induced delay

In an ideal scheduler implementation, each time the scheduler selects the next packet to send, it will always select the priority queue if this contains a waiting packet. But this does not mean the priority queue will always be served immediately a new voice packet arrives, because the scheduler may already be in the process of serving a data packet. Even in an ideal implementation, the worst-case delay is where a maximum-size data packet has just started to be served, and completion will take a time equal to the data-packet size divided by the link-speed. In practice though, scheduler implementations are not ideal, and in some cases the scheduler includes an output buffer that may lead to two or possibly even more data packets being served ahead of a recently arrived voice packet. This results in voice packets experiencing a variable delay, known as jitter.

The importance of this delay component depends on link-speed. For example, a 1500 byte packet has a transmission time of about 6 ms on a 2 Mbit/s link, which may be considered relatively small. But on a 256 kbit/s link, the transmission time for a 1500 byte packet is 50 ms, which is likely to cause unacceptable jitter. To overcome this problem, a technique called fragmentation may be employed on low-speed links. This involves chopping up data packets into small fragments, and allows voice packets to be transmitted between fragments instead of having to wait for an entire data packet to be sent. The fragmentation process is local to the particular link on which it is configured, and is essentially invisible to the end-to-end data traffic being carried. For example, using a 300 byte fragment-size on a 256 kbit/s link reduces this jitter component by a factor of five.

A2.2 Voice contention delay

This delay component is caused when more than one simultaneous VoIP session is active, leading to possible contention within the voice queue itself. If N sessions are active, the worst-case is when a packet from each session arrives simultaneously, leading to the need to queue the last $N-1$ packets. Statistical considerations must be applied to analysing this delay component though, which should therefore be defined in probabilistic rather than absolute terms, leading to a delay bound that will be exceeded only with some very small probability. In characterising this, a well-known statistical process may be applied, referred to as the $N{*}D/D/1$ process [7], which under some conditions may be approximated by the much more tractable $M/D/1$ process [8].

In short, such analysis shows that contention delay increases non-linearly, as link-speeds are reduced, and as the proportion of link bandwidth assigned to voice is increased. This leads to the need for some sensible design rules for the maximum proportion of voice traffic that may be carried, especially on links of bandwidth less than about 2 Mbit/s. Voice packet size is also a factor here, with larger voice packet sizes producing proportionately larger queuing delays, where voice packet size is a function of the chosen codec and framing-rate.

A3 Summary

In conclusion, the tight performance constraints of VoIP may be realised through the application of DiffServ principles, allied to a well-dimensioned core network, sensible constraints on the maximum level of voice traffic, and use of fragmentation on low-speed access tails.

References

1 Willis P J: 'An Introduction to quality of service', BT Technol J, 23, No 2, pp 13—27 (April 2005).

2 Blake S et al: 'An architecture for Differentiated Services', IETF RFC2475 (December 1998).

3 Davie B et al: 'An Expedited Forwarding PHB (per-hop behaviour)', IETF RFC3246 (March 2002).

4 Heinanen J et al: 'Assured Forwarding PHB Group', IETF RFC2597 (June 1999).

5 Floyd S and Jacobson V: 'Random-detection Gateways for Congestion Avoidance', IEEE/ACM Transactions on Networking, 1, No 4, pp 397—413 (August 1993).

6 Dann T et al: 'The applications-assured infrastructure', BT Technol J, 23, No 2, pp 73—80 (April 2005).

7 Eckberg A E: 'The single server queue with periodic arrival process and deterministic service time', IEEE Trans Commun, 27, pp 556—562 (March 1979).

8 Schormans J A and Pitts J M: 'From Erlangs to Excess Rate', Journal of the IBTE, 2, part 4, pp 54—60 (December 2002).

Simon Carter studied at Reading University, where he gained a BSc in Physics, and a PhD for studies in electronics.

He joined BT in 1987, initially working on long-haul optical transmission systems, before moving to the Research Department to work on a range of optical and IP network projects. This was followed by a spell in performance engineering, supporting a range of BT's deployed IP networks.

He is currently a network designer for the MPLS platform, specialising in quality of service, and was the lead designer for the recent DSCP CoS development. More recently he has also been involved in BT's 21st century network architecture programme.