ELSEVIER

Computer Networks 40 (2002) 131–148

COMPUTER NETWORKS

www.elsevier.com/locate/comnet

Engineering a multiservice IP backbone to support tight SLAs

Clarence Filsfils [a,*], John Evans [b]

[a] *Cisco Systems, Brussels, Belgium*

[b] *Cisco Systems Ltd., 10 New Square Park, Bedfont Lakes, Feltham, Middlesex, UK W14 8HA*

Abstract

This paper describes technologies that enable IP service providers to offer tighter service level agreements for IP performance, in order to create competitive advantage and better serve their customers. The SLA parameters that need to be tightened are defined and then the technologies that should be considered are described, together with the decision criteria on where each technology should be used. This paper is based upon current best practise and includes results from both lab testing and deployment experience. The specific technologies discussed are differentiated services, fast IGP convergence, and traffic engineering. Consideration is given to how these technologies should be deployed and operated.

IDT: Differentiated services; MPLS; Traffic engineering; IGP convergence

1. Introduction

When using public transport, the traveller may benefit from contractual commitments from the transport service provider, for example that 95% of journeys will arrive within 5 min of the scheduled time. The commitments may include other parameters or metrics such as number of stops en route and any meals included. The more competitive the market for the particular service, the more comprehensive and the tighter the commitments or service level agreements that are offered. In the same way, the increase of competition between IP service providers (SPs) together with the heightened importance of IP to business operations has led to an increased demand and consequent supply of IP services with tighter service level agreements (SLAs) for IP performance.

The IP technical community has developed a set of technologies that enable IP networks to be engineered to support tight SLA commitments:

- *Differentiated services (Diffserv).* The Diffserv architecture allows differentiated delay, jitter and loss commitments to be supported on the same IP backbone for different types or classes of service.
- *Faster IGP convergence.* New developments in Interior Gateway routing protocols (IGPs) allow for faster convergence upon link or node failure, hence enabling higher service availability to be offered.
- *MPLS traffic engineering.* MPLS traffic engineering (Diffserv-aware or not) introduces constraint-based routing and admission control to

* Corresponding author.

E-mail addresses: cfilsfil@cisco.com, cf@cisco.com (C. Filsfils), joevans@cisco.com (J. Evans).

1389-1286/02/$ - see front matter
PII: S1389-1286(02)00271-2

IP backbones. This allows optimum use to be made of the installed backbone bandwidth capacity, or conversely allows the same level of service to be offered for less capacity. It can also be used to ensure that the amount of low-jitter traffic per link does not exceed a specified maximum.

- *MPLS traffic engineering fast reroute.* MPLS traffic engineering fast reroute is an IP protection technique that enables connectivity to be restored around link and node failures in a few tens of milliseconds.

For an SP IP service, the SLA commitments are based on delay, jitter, packet loss rate, throughput and availability. This paper focuses on defining these SLA parameters, and describing why and how the above technologies should be used in order for these SLA parameters to be tightened. Consideration is also given to how networks using these technologies should be operated. There are other factors that will affect a service provider's ability to offer tight SLAs in addition to the technologies described, but which are not covered in this paper; these include: network security and BGP convergence.

In focussing on service provider IP backbone networks, it is noted that the mechanisms employed at the edge of the network to deliver tight SLAs may be different from those used in the core. In the backbone, where traffic is aggregated, SLA requirements for a traffic class can be translated into the appropriate bandwidth requirements, and the problem of SLA assurance can effectively be reduced to that of bandwidth provisioning. At the network edge, other considerations, such as serialisation delay, become significant. The mechanisms employed at the edge of the network are not considered further in this paper.

2. Customer requirements—what an SLA should commit to

The following metrics are highlighted as the most important for specifying the quality of an IP backbone service [15], For each metric, typical values are given for different types or classes of service.

2.1. Network one-way delay

In IP backbone terms, network one-way delay characterizes the time difference between the reception of an IP packet at an ingress point of presence (POP) and its transmission at an egress POP. Network one-way delay is made up of four components:

- *Propagation delay.* Propagation delay is constrained by the speed of light in a medium and for optical fibre is around 5 ms per 1000 km. Propagation delay can vary as network topology changes, when a link fails, for example, or when an underlying network (e.g. SDH/SONET) reroutes its circuit paths.
- *Switching delay.* Switching or processing delay is the time difference between receiving a packet on an incoming router interface and the enqueuing of the packet in the scheduler of its outbound interface. Switching delays on today's high-performance routers are negligible, typically in the order of 10–20 μs per packet.
- *Scheduling delay.* Scheduling (or queuing) delay is defined as the time difference between the enqueuing of a packet on the outbound interface scheduler, and the start of clocking the packet onto the outbound link. This is a function of the scheduling algorithm used and of the scheduler queue utilization, which is in turn a function of the queue capacity and the offered traffic load and profile. This effect is analysed in more detail later in this paper.
- *Serialisation delay.* Serialisation delay is the time taken to clock a packet onto a link and is dependent upon the link speed and the packet size. Serialisation delay is considered negligible at link speeds above STM-1/OC3, such as backbone links: a 1500 byte packet is clocked at STM-1/OC3 rate (155 Mbps) in 80 μs, at STM-16/OC48 rate (2.5 Gbps) in 5 μs and at STM-64/OC192 rate (10 Gbps) in 1.25 μs.

The goal commonly used in designing networks to support voice over IP (VoIP) is a delay budget

of 150 ms from mouth to ear. [1] A design should apportion this budget to the various components of network delay (propagation delay through the backbone, scheduling delay due to congestion, and the access link serialisation delay) and service delay (due to VoIP gateway codec and de-jitter buffer). Propagation delay is often budgeted by using the widest diameter in the network, which for example, in a national network in the US would give a worst case (coast-to-coast) of 6000 km or 30 ms of one-way propagation delay.

Interactive data applications often require a round trip time (RTT) of less than 250 ms to allow for a smooth interaction between the human and the application server.

SPs today typically commit on monthly average one-way delay over all their POP-to-POP pairs.

2.2. Network delay-jitter

Network delay-jitter characterizes the variation of network delay; it is generally computed as the variation of the delay for two consecutive packets. Jitter is caused by the variation in the components of delay previously described:

- *Propagation delay.* Propagation delay can vary as network topology changes, when a link fails, for example, or when the topology of a lower layer network (e.g. SDH/SONET) changes, causing a sudden peak of jitter. Current IP backbone experience would suggest that these occurrences are more common than is generally believed [6].
- *Switching delay.* Switching delay can vary as some pacakets might require more processing than others. This effect is becoming less of a consideration as packet switching is implemented using hardware pipelines whose switching delay characteristics are deterministic.
- *Scheduling delay.* Variation in scheduling delay is caused as scheduling queues oscillate between empty and full.

Jitters buffers (also known as play-out buffers) are used to remove delay variation by turning variable network delays into constant delays at the destination end systems. Consequently, in networks that are engineered to support low-delay services such as VoIP it is important that they are also engineered for low jitter. IP backbones that are engineered to support high-quality VoIP services typically budget for 5–10 ms of jitter in the backbone; assuming 10 backbone hops, this gives a jitter budget per hop of 500–1000 μs. Schedulers that implement a priority queuing mechanism such as the "modified" [2] Deficit round robin (mDRR) scheduler [25] implemented on high-performance routers have jitter characteristics 5–10 times better than this. Data applications do not generally require specific constraints on jitter.

2.3. Loss

Loss characterizes the packet drops that occur between the ingress link of the ingress POP and the egress link of the egress POP. It is observed that US backbone service providers usually offer an average monthly loss rate of less than 1%. Backbones engineered for high-quality VoIP services typically plan for a loss rate of less than 0.25%. The same target range is also used for high-quality data services; TCP throughput has been shown to decrease as an inverse of the square root of the probability of packet loss [19].

2.4. Bandwidth and throughput

IP services are commonly sold with a defined bandwidth, where the bandwidth reflects the access link capacity provisioned for the service. Defined bandwidth may not be the same as achieved throughput, however. Throughput characterises the available user bandwidth between an ingress POP and egress POP. The requirement for this SLA parameter is obvious for point-to-point services such as virtual wires, as being defined by the Pseudo Wire Emulation Edge to Edge (PWE3)

[1] ITU standard G.114 states that 150 ms of end-to-end one-way delay does not cause a perceivable degradation in voice quality for most use of telephony.

[2] The DRR algorithm used on Cisco routers has been modified by Cisco to add support for a strict priority queue.

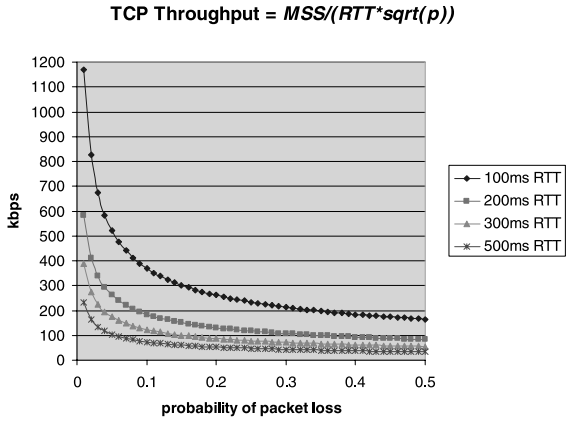

Fig. 1. TCP throughput as a function of packet loss and RTT with MSS = 1460 bytes.

Working Group [22] within the IETF. For multipoint-to-multipoint services, such as VPNs, the SLA definition will need to exclude cases where the loss of throughput is due to customer-based aggregation. For example, where 10 sites each with STM-1/OC3 access links are all sending traffic at full rate to a single site with only an STM-4/OC12 access link [9].

For TCP/IP traffic, achieved throughput is largely dependent upon the probability of packet loss and the achieved RTT. Consequently, contracted bandwidth may not relate to achieved throughput. The graph in Fig. 1 uses the relationship defined by Mathis [19] to show how TCP throughput varies as a function of packet loss and RTT.

2.5. Per-flow sequence preservation

Per-flow sequence preservation is not yet a common component of IP service SLA commitments even though it is accepted best practise in IP network design. Due to the impact that packet reordering can have on some applications, however, it is starting to be offered by SPs for high-quality data and video services. To prevent out of order packet delivery, it is important that any load balancing is achieved at a per-flow level rather than at a per packet level, and scheduling algorithms assure that packets from the same flow are serviced in order and from the same queue.

Real-time video is commonly impacted by reordering as the end-user applications either do not have the time to reorder the received frames or simply do not support this functionality; in both cases, reordering means a higher packet loss. Most TCP implementations interpret the receipt of three consecutive similar acknowledgements as an indication of packet loss and hence retransmit the next packet and slow down their rate; reordering therefore has a significant impact on TCP throughput [18]. Whilst the reordering magnitude would need to be very significant in order to affect a VoIP flow whose inter-packet gap is a multiple of 10 ms, the reordering might cause significant jitter due to variation in the propagation delay (for example, US east coast to west coast via a north or the south route).

2.6. Availability

Availability for IP services is generally defined in one of two ways; in terms of either network availability or service availability. Network availability is defined as the fraction of time that network connectivity is available between a specified ingress point and a specified egress point. Service availability is defined as the fraction of time the service is available between a specified ingress point and a specified egress point within the bounds of the defined SLAs. There can be overlap between the definition of network or service availability and the definition of other SLA parameters, for example, 100% network availability implies 0% packet loss.

2.7. Classes of service

Amongst the QoS enabled IP services offered by service providers today there is common support for classes of service designed to meet the needs of three traffic types: real-time traffic, business data traffic and best-effort data traffic. Typical SLA commitments and service characteristics for these classes are as follows:

- *Real-time.* This class targets applications such as VoIP and video. It is defined in terms of low loss (less than 0.25%), low delay, and low jitter (typ-

ically 5 ms within the backbone) and has a specified bandwidth and availability. Attainable throughput is derived from bandwidth and loss. The class may support a commitment for per-flow sequence preservation.

- *Business data.* This class represents business critical interactive applications such as SNA, SAP R/3, Telnet, and possibly intranet Web applications to selected URLs. It is defined in terms of delay (RTT should be less than 250 ms), and loss (less than 1% loss rate is typical, with targets of less than 0.1% also available), with a specified bandwidth and availability. Attainable throughput is derived from loss and RTT. Jitter is not important for this service class and is not defined. The class may support a commitment for per-flow sequence preservation.
- *Best-effort data.* This class represents all other traffic that has not been classified as *real-time* or *business*. It is defined in terms of a loss rate with a specified bandwidth and availability. Attainable throughput is derived from loss. Delay and jitter are not important for this service and are not defined. The class may support a commitment for per-flow sequence preservation.

2.8. Design objectives

In designing an IP backbone network with support for different classes of traffic, service providers have three key design objectives:

- Committing to the different per class SLA requirements.
- Making optimal use of available bandwidth.
- Keeping the design as simple as possible.

In the remainder of this paper, we describe the different technologies that are required to satisfy these requirements.

3. The differentiated services architecture

3.1. IP backbone diffserv overview

The Diffserv architecture [3] is the preferred technology for large-scale IP QoS deployments today, such as service provider backbone networks. Diffserv achieves scalability through performing complex QoS functions such as classification, marking, and conditioning operations at the edges of the network. Traffic is classified and then marked using the Diffserv code point (DSCP) [21] into a limited number of traffic aggregates or classes. Within the core of the network, scheduling and queuing control mechanisms are applied to the traffic classes based upon the DS field marking; all traffic conditioning and dropping is handled intelligently at the network layer using IP Diffserv quality of service mechanisms. Diffserv is not prescriptive in defining the scheduling and queuing control algorithms that should be implemented at each hop, but rather, uses a level of abstraction in defining the externally observable forwarding behaviours, termed per-hop behaviours (PHBs), that can be applied to traffic at each hop. Currently, three PHBs are defined:

- *The expedited forwarding (EF) PHB.* The EF PHB [8,16] is used to support traffic with low loss, low delay, low jitter, assured bandwidth requirements, such as VoIP.
- *The assured forwarding (AF) PHB.* The AF PHB [14] is used to support data traffic with assured bandwidth requirements.
- *The default PHB.* This PHB [21] represents the default forwarding behaviour. Packets, which are not identified as belonging to another class, belong to this aggregate.

A typical Diffserv backbone implementation, that could be used to support the three classes of service previously defined, is described as follows:

- *Real-time.* The real-time traffic class is treated with an EF per-hop behaviour. This is typically implemented using a strict priority queue within a scheduling mechanism, [3] which services the strict priority queue with priority above all other queues. Such a queuing mechanism assures that the real-time class is isolated from the impact of the other classes.

[3] Such as the modified Deficit Round Robin scheduler used on high-performance Cisco routers.

The forwarding resources for the class are allocated to ensure that with the expected load there will be no congestion within the class, such that it receives good deterministic service with low loss, low delay and low jitter.

- *Business data.* Business data class traffic is treated with an AF per-hop behaviour. This is typically allocated to a queue within a weighted-fair-queuing-like scheduling mechanism. Such a queuing mechanism assures that the class receives a bandwidth allocation relative to the best-effort data class and that it can use all available forwarding resources after the VoIP and best-effort data class queues have been serviced.

  The forwarding resources for the class are allocated to ensure that for the majority of the time, with the expected load there is no congestion within the class such that it receives good deterministic service with low loss.

  The assumption is made that the majority of the business data class traffic is TCP/IP [20], and hence the Random Early Detection (RED) [13] congestion control mechanism is used within the class queue rather than tail drop to ensure that TCP throughput within the class is maximised when congestion occurs. If a service is defined with in-and out-of-contract capabilities (similar to the concept of discard eligibility [DE] set/unset within frame-relay), Weighted RED (WRED) is used to have two different RED profiles [4] per queue: an aggressive profile for out-of-contract traffic, and a gentler profile for in-contract traffic to make sure that in case of class congestion, out-of-contract packets are dropped first. [5]

- *Best-effort data.* The best-effort class traffic uses an AF per-hop behaviour. This would typically be allocated to a queue within a weighted fair-queuing scheduling mechanism. Such a queuing mechanism assures that the class receives a bandwidth allocation relative to the business data class and that it will be able to use all available forwarding resources after the VoIP and business data class queues have been serviced.

  As with the business data class, the assumption is made that the majority of the best-effort data class traffic is TCP/IP, therefore the RED congestion control mechanism is used within the class queue rather than tail drop to ensure that TCP goodput within the class is maximised when congestion occurs.

This paper does not consider the classification, conditioning and marking functions performed at the edges of the network, but assumes that such mechanisms are used. An edge design may offer more classes of service; for example, two or three different classes for business data traffic to either isolate applications and/or ease the budgeting of the bandwidth cost between different customer departments. In this case, the capability to map several edge classes into a single aggregate backbone class is required on the SP's edge router. Several edge business data classes may be mapped into the backbone business data class or two distinct edge VoIP and video classes may be mapped into the aggregate backbone real-time class. This mapping can be realized in two ways:

- A backbone class can match several DSCPs. For example, if at the edge of the network, a DSCP value of 40 represents VoIP at the edge and a value 32 represents Video, the backbone aggregate real-time class would match both DSCP 40 and DSCP 32.
- When MPLS is used in the backbone, the edge SP router can set the 3 bit MPLS experimental (EXP) field as a function of the received DSCP. For example, if an EXP value of 5 is used for the aggregate real-time backbone class, the SP's edge router will impose MPLS labels headers with an EXP value of 5 for packets received

[4] The characteristics of RED are defined by a minimum queue threshold (min_{th}), maximum queue threshold (max_{th}), and probability of discard at $max_{th}(max_p)$. WRED allows multiple red profiles to be supported in the same queue with separately defined min_{th}, max_{th} and max_p per profile. This results in different drop characteristics (and consequently probability of drop) per profile.

[5] This is achieved by choosing a maximum threshold for out-of-contract traffic smaller or equal to the minimum threshold for in-contract traffic.

with DSCP 40, representing the edge VoIP class, or with DSCP 32, representing the edge Video class.

3.2. Diffserv backbone design for low loss and low jitter

Designing an IP backbone network for low loss, low delay and low jitter can be relatively simple: one simply needs to overprovision the bandwidth compared to the actual load [4,6,7]. Ref. [6] shows that for a best-effort IP backbone, worst case jitter was measured at less than 1 ms for probes sent at 1Mbps during a seven-day period between the east and west coast POPs of a Tier-1 Internet Service Provider (ISP). During this period, the loss rate was zero. These measurements demonstrate the excellent performance that can be achieved in an IP backbone when designed with high-speed links (from STM-4/OC12 to STM-64/OC192) and with conservative capacity provisioning rules where links are upgraded when utilisation reaches 40–50%, aiming to ensure that there is at least twice as much capacity as actual load.

Such a simple design rule allows tight SLAs to be achieved for delay, jitter and loss. Unfortunately, however, this does not satisfy our initial requirements as this represents an expensive option for the SP. If, for example, between two POPs there is 150 Mbps of VoIP traffic and 1.5 Gbps of best-effort data traffic, then by the above rule, twice the sum of the VoIP and best-effort traffic would be needed to assure low jitter and loss of the VoIP class. In this case, the sum of the VoIP and data traffic is 1.65 Gbps and hence 3.3 Gbps of capacity need be provisioned. In practice, this would typically be supported using two STM-16/OC48 links, resulting in 5 Gbps of bandwidth being deployed to support 1.65 Gbps of aggregate load with 150 Mbps of low delay, low jitter, low loss traffic.

Diffserv provides a solution to this problem, in that it allows per class virtual backbones to be built on a single physical backbone. This gives SPs the flexibility to have different under- or over-booking ratios (the ratio of offered load to available capacity) for each service class. Using the previous example, this could allow the VoIP class capacity to be over provisioned by a factor of 4 [6] hence ensuring that the class receives good service (with low delay, low jitter and low loss), whilst the data class capacity could be over provisioned by a factor of 1.2 (a realistic figure still giving good service). This would result in 2.1 Gbps of bandwidth required in total, or rounded up to a single STM-16/OC48 link, which represents a potential saving of 1× STM-16/OC48 link over the non-Diffserv case. To explain how this calculation was derived, the VoIP class traffic is assumed to be serviced from a strict priority queue and thus will effectively have access to all bandwidth on the physical link; for an STM-16/OC48 link this would result in effective over provisioning by a factor of 2.5 Gbps/150 Mbps $= \sim 16$. The best-effort class traffic being serviced from a weighted-fair queue would, however, have access to all available bandwidth on the physical link once the VoIP class traffic has been serviced; for an STM-16/OC48 link this would result in effective bandwidth over provisioning by a factor of [(2.5 Gbps–150 Mbps)/1.5 Gbps] $= \sim 1.6$.

This example is intended to highlight two key points: that backbone Diffserv deployment is conceptually simple and that the concepts involved have been proven by experience with deployed best-effort IP networks. It is simple in that it allows different service levels to be supported merely by using different under- or over-provisioning ratios per class; the higher the available capacity compared to the offered load, the tighter the SLA (lower delay, jitter, and loss rates) that can be supported. [7] It is proven in that empirical evidence

[6] In practise, an overprovision factor of 4 is used, based upon the simple assumption that single-link or node failure conditions can result in a doubling of the per link load. Consequently, using as overprovision factor of 2*2* the maximum expected load in non-failure conditions aims to ensure that even in failure conditions, there is twice as much capacity as traffic load and hence low delay, jitter and loss service is maintained.

[7] Assuming two classes, A and B, such that A's overprovisioning ratio is higher than B. A tighter delay SLA for A can be expressed either by a smaller delay bound for A than B but with the same availability, or by the same delay bound but with an higher availability for A than for B, or a combination of both schemes.

from high-speed best-effort (single class of service) SP IP backbones indicates that extremely good jitter and loss targets can be achieved with simple rules of over provisioning.

3.3. Diffserv backbone deployment considerations

Consideration obviously needs to be given to whether the cost of deploying Diffserv outweighs the benefits it provides. There is no generic answer to this question, and the benefits that will be gained will vary deployment by deployment. In the example above, if the Diffserv deployment cost exceeds the cost of an additional STM-16/OC48 link (and the router ports which terminate it) then there is clearly no sense in deploying Diffserv. We consider some of the most significant factors that impact the economic viability of deploying Diffserv:

- *Economic benefit*. The maximum potential economic benefit stands to be gained from Diffserv deployments where the traffic requiring the highest SLA targets represents a minor proportion of the overall capacity. As the previous example demonstrates, the absence of Diffserv leads the designer to provision capacity equal to the aggregate load across all classes multiplied by the over dimensioning ratio of the tightest-SLA class. This can be extremely expensive when the tightest-SLA class represents a low proportion of the aggregate traffic. Conversely, when all classes require the same level of service, and hence the same overbooking ratio, there is no benefit to be gained from Diffserv.
- *Impact on router performance.* If Diffserv EF/AF forwarding behaviours have an impact on router forwarding performance, the less aggregate throughput the router can support with Diffserv enabled, and consequently, the higher the per port cost of the network deployment. Today's high-performance routers typically implement the EF/AF forwarding behaviours in ASICs, ensuring that there is no forwarding penalty associated with the support of the Diffserv functionality
- *Simplicity of deployment.* Backbone Diffserv deployments generally require relatively minor and simple changes to existing router configurations. A typical Diffserv design consists in defining three queues: an EF class queue for real-time traffic such as VoIP, and two AF class queues, one for the business data class and the other for the best-effort class. The real-time class traffic is serviced with a priority queuing treatment, and the expected load for the class is below 25% (using an overprovision factor of 4) of the available link capacity in most designs. The bandwidth remaining once the real-time class has been serviced is allocated with 90% to the business data class and 10% to the best-effort class. An example router configuration, which could be used to implement the described Diffserv design, is shown in Fig. 2; additional lines of configuration required to implement the Diffserv policy are shown in bold type in the figure.

  As can be seen from Fig. 2, only 12 additional lines of configuration are required to implement the Diffserv policy for a single interface. Only a single additional line of configuration is required for each additional interface where this Diffserv template will be configured. Typically, in backbone Diffserv deployments these configurations are applied once, and then remain static and are never changed.

  Furthermore, migrating a backbone to Diffserv can be achieved seamlessly: the backbone configuration can be undertaken independently of the configuration required at the network edge to ensure that traffic is appropriately conditioned and marked on ingress to the network. The benefits of Diffserv, however, will not be realised until both edge and backbone components are complete.
- *NMS/OSS requirements.* Backbone network management systems (NMS) and operational support systems (OSS) typically need enhancing to support Diffserv deployments:
  - The NMS system needs to be enhanced to retrieve bytes/packets transmitted and dropped per class rather than per interface. In response to the Diffserv deployments that have occurred during the 24 months preceding the publication of this paper, NMS applications now typically provide support for such statistics.

```
                                          This section defines the classification criteria used
class-map REAL_TIME                       to determine which traffic goes in which queues.
  match ip dscp 40                        In this case traffic marked DSCP 40 and 32 will be
class-map BUSINESS                        classified as Real-time and Business Data classes
  match ip dscp 32                        respectively, whilst all other traffic will be
class-map BEST_EFFORT                     classified as the Best-effort class
  match any
!
                                          ------------------------------------------------------
policy-map DIFFSERV_POLICY                This section is a template, which defines the actual
  class REAL_TIME                         queuing treatment that each class will receive. A
    priority                              strict priority queue is defined for the Real-time
  class BUSINESS                          class, whilst the Business data and Best-effort
    bandwidth remaining percent 90        classes are allocated to queues, which are
  class BEST_EFFORT                       guaranteed a minimum of 90% and 10% of the
    bandwidth remaining percent 10        remaining bandwidth respectively. The RED
!                                         configuration for the data classes is not shown.
!
                                          ------------------------------------------------------
interface pos 1/0                         In this section, the queuing template defined above
  service-policy outbound DIFFSERV_POLICY is attached to an actually interface
```

Fig. 2. Example backbone router Diffserv configuration.

- The deployment of an active SLA probing system is highly advisable [15] in order to be able to monitor (and report) delay and jitter. Some router vendors implement software agents in their routers [8] that send and receive probes with user-definable DSCP and protocol identities (e.g. FTP, HTTP, DNS). Leveraging the installed base of routers in each POP allows rapid deployment of an SLA active monitoring system without any major rollout of new network equipment.

- *Capacity planning.* In terms of operational process, the capacity planning of a Diffserv backbone is similar to a single-class best-effort IP network: load statistics are collected on a per-class of service basis and when load thresholds are reached, an addition of network bandwidth is triggered. The accuracy of this capacity planning can be tuned based upon the active SLA probing results which allows correlation between per class load and SLA parameter reports of delay and jitter.

- *Training*. Diffserv is a new technology and therefore it is inevitable that training of design and operational staff will be required to support a backbone Diffserv deployment.

3.4. Diffserv backbone performance characteristics

We conclude the discussion on Diffserv by presenting the results of router based testing, which illustrate both the tight-latency, jitter, and loss capabilities of today's router technology and the potential benefits and characteristics that can be achieved with backbone Diffserv deployment described above.

The testing was undertaken using a Cisco 12416 router. This router has a distributed architecture that supports the EF and AF per-hop behaviours implemented in ASICs on each line card using a deficit round robin (DRR) [25] scheduling algorithm, which has been modified by Cisco to add support for a strict priority queue for EF class traffic.

The Diffserv configuration shown in Fig. 2 was used for all tests. The packet size used for real-time traffic during testing was 200 bytes, whilst the business data and best-effort traffic followed an

[8] For example, service assurance agent (SAA) functionality in Cisco IOS.

Internet-mix packet size distribution. [9] The router under test has three ingress STM-16/OC-48 ports receiving traffic from a traffic generator. The router aggregates this traffic and forwards it onto the single-hop link under test which is an STM-16/OC-48 Packet over SDH/SONET (POS) link.

Three characteristics of the router EF and AF implementation, which are key to the successful deployment of the Diffserv design described above, form the basis of the tests and results presented:

- *Latency of EF class.* The first test measured the one-way delay of the real-time class in the presence of interface congestion. A worst-case delay success criterion of 500 μs was set in order to ensure that the 500 μs delay-jitter target was never exceeded. [10]
- *Latency of AF class.* The second test measured the delay of the business data class traffic under increasing load within that class.
- *Accuracy of AF bandwidth allocation.* The final test measured the bandwidth allocation accuracy of the AF class traffic (business data and best-effort), for different relative bandwidth allocations. Successful Diffserv deployment depends upon being able to manage the relative under- and over-booking ratios between the classes, which is in turn dependent upon the accuracy of the scheduler implementation in terms of AF bandwidth allocation.

3.4.1. Latency for EF traffic

The first results demonstrate the low delay that can be achieved using an EF compliant priority queue scheduler. Fig. 3 charts the recorded percentile distribution for the real-time class (EF) traffic delay through a single-hop link running at STM-16/OC48 and being congested with 30% of VoIP, 45% of business traffic and 150% of best-effort traffic.

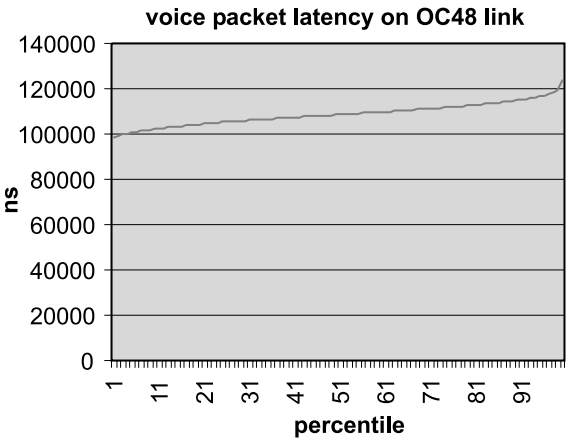

Fig. 3. Latency for EF class.

Fig. 3 clearly illustrates the low-jitter service provided by the priority queuing mechanisms to the VoIP (EF) traffic. The percentile 100 delay of the VoIP packets is 125 μs. This result was independent of the load of the other classes even under 225% congestion of the outbound port. These results are significantly better than the target of 500 μs.

3.4.2. Latency for business class

This test demonstrates that very good maximum delay can be achieved for a well-provisioned AF class queue, such as the business data class queue in our case. In this test, the link under test was loaded with bursty traffic in the PQ up to 30% of the STM-16/OC48 link rate, and 150% of best-effort class traffic. The business data class load was then varied from 0% to 200% (of the configured business class capacity) and the maximum delay of the business class traffic was measured. Fig. 4 illustrates the test results.

With business class load less than business class capacity, no packet loss is experienced for the class and the latency remains extremely low at ~160 μs until ~85% business class load. As the business load increases to 100% the latency increases to ~1 ms at 100% load. Above 100% business class load packet loss occurs and the average delay measured increases to ~100 ms at 220% load, which in the case of our testing was the expected result

[9] Fifty eight percentage of the packets are small packets (40 bytes), 33% of the packets are medium sized packets (552 bytes) and 9% of the packets are large packets (1500 bytes).

[10] The delay-jitter can never exceed the difference between the best-case and worst-case measured one-way delay.

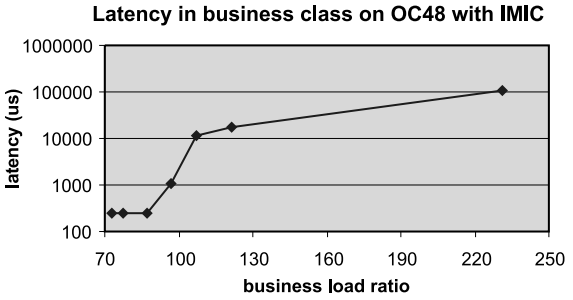

Fig. 4. Latency for an AF class.

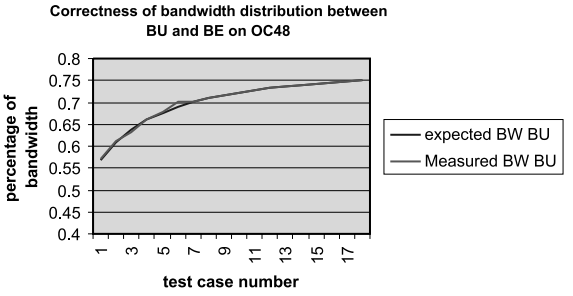

Fig. 5. Bandwidth accuracy for AF queues.

determined by the RED configuration that we used. [11]

3.4.3. Bandwidth allocation between AF queues

The final test demonstrates the accuracy of the bandwidth allocation that can be achieved between different AF classes. In this test, the link under test was loaded with bursty traffic in the PQ up to 30% of the STM-16/OC48 link rate, and variable equal rates of business and best-effort class traffic giving an aggregate interface load of greater than 200% of the STM-16/OC48 link rate. The ratio of the configured bandwidth allocations between the business and best-effort classes was then varied, and the actual bandwidth allocation ratio was measured and compared to the expected result. Fig. 5 graphs the difference between the measured business bandwidth allocation and the expected bandwidth allocation, as a percentage of the STM-16/OC48 link rate.

Fig. 5 shows different test cases, which correspond to different relative bandwidth allocations between the business and best-effort classes (50/50, 66/34, 75/25 etc.). In all tests, the bandwidth allocation accuracy is well within 1% of the expected result.

4. Fast IGP convergence

Link or node failures in an IP backbone cause packet losses until the network has reconverged around the failed link or node. These packet losses directly impact the availability that can be offered for SLAs across all classes. To assess the significance of this one can compute the amount of downtime corresponding to different availability commitments: the often-quoted 99.999% or "five-nines"-target figure for network availability potentially equates to less than 1 s of downtime per day.

Another way to illustrate the significance of downtime due to a link or node failure is to consider the impact on applications and end-users. For VoIP calls, the end-users will perceive a glitch in the call as soon as a few samples are lost. For example, with one sample every 20 ms, a loss in connectivity of 100–150 ms will be perceptible to the human ear. If the connectivity loss is for 1–2 s the call can be dropped. Consequently, networks supporting high-quality VoIP services are being engineered for network convergence upon link/node failure of 1 s or less. The time taken for an IP network to reconverge is dependent upon the size of the network, [12] the Interior Gateway Routing protocol (IGP) used and its specific configuration. For high-SLA availability targets to be offered, it is important that the routing protocol is tuned for rapid convergence. A key component of tuning IGP convergence is the tuning of the timers, which determine how frequently the main routing protocol events can occur. Historically, this has resulted in a trade-off between rapid convergence

[11] The worst-case latency through a RED-controlled queue is related to the $\max_{th}$. In our design, this is tuned to the *pipesize* (RTT* bandwidth) with an estimated RTT of 100 ms.

[12] In terms of number of links and nodes in the network and number of routes carried in the IGP.

and increased routing protocol stability: short timers resulted in rapid convergence but with more potential for instability, where longer timers resulted in increased stability but slower convergence. The pragmatic result of this trade-off was that routing protocol timers were generally set conservatively and IP network convergence was typically a few tens of seconds.

Experience gained from large-scale service provider deployment, however, indicated that such IGP implementations were very stable and hence that more emphasis could be placed on faster convergence. Further, recent developments to IS–IS and OSPF link state IP IGPs have focussed on combining the best of both worlds leading to significant reductions in the convergence that can be achieved whilst still maintaining stability. Whereas previously IGP timers were static and long, now with the introduction of dynamic timers they can adapt their responsiveness depending upon the stability of the network. This allows IGPs to be tuned such that when the network is stable, their timers will be short and they will react within a few milliseconds to any network topology changes. In times of network instability (e.g. caused by a flapping link), however, the IGP timers will increase in order to throttle the rate of response to network events. This scheme ensures fast convergence when the network is stable and moderate routing protocol overhead (e.g. CPU cycles consumed) when the network is unstable.

In addition to the advancements in the tuning of routing protocol timers, a number of other developments have improved the IGP convergence that can be achieved:

- *Partial routing calculation and incremental SPF* [23,24]. IGP implementations have been enhanced to react more optimally to various topology changes. For example, with the introduction of partial routing calculations for IS–IS and OSPF, if only an IP leaf subnetwork changes without a topological change to the shortest path tree, then the router does not need to recompute the shortest-path tree, it just recomputes its routing table based upon the existing tree. On the other hand, when only a part of the graph has changed, incremental SPF optimises the calculation by only recomputing the part that has changed instead of the whole graph. Such optimisation leads to faster routing computations, hence to faster convergence. Early empirical data from large-scale network deployments suggests that the average IGP routing table computation time can be reduced by 90% due to such optimisations.
- *Design practice*. Best practise for service provider IGP design today aims to reduce the number of routes that are carried in the IGP; all Internet and customer routes are carried by the Border Gateway protocol (BGP); in some cases even the interior link prefixes are carried in BGP. This practise significantly reduces IGP routing table computation times and hence results in faster IGP convergence. Obviously, the routing table for a network of 100 routers and 1000 IP prefixes can be computed faster than one with 100 routers and 10,000 prefixes.
- *Rapid layer 2 failure detection.* Most service provider backbones today are built using Packet Over SDH/SONET (POS) links where signalling inherent in SDH/SONET is designed to detect link or node failures in less than 10 ms. This rapid failure detection can considerably speed up convergence, when compared to other media such as Ethernet, which rely on hellos at the IP layer to detect failure.

The combination of these optimisations has resulted in a reduction of IGP convergence times from several 10 s of seconds, to 1–2 s being pragmatically achievable today. This has been confirmed in testing in a 1000 router network with IP 4000 prefixes in the IGP (work in progress). Further, as we learn from additional deployment experience, and with additional tuning and enhancements, sub-second IGP convergence may become a realistic possibility.

5. MPLS traffic engineering and diffserv-aware traffic engineering

5.1. MPLS traffic engineering

In conventional service provider IP networks routing protocols such as OSPF and IS–IS for-

ward IP packets on the shortest cost path to the destination IP address of each IP packet. The computation of the shortest cost path is based upon a simple additive metric, where each link has an applied metric, and the cost for a path is the sum of the link metrics in the path. Availability of network resources, such as bandwidth, is not taken into account and, consequently, traffic can aggregate on the shortest path, potentially causing links on the shortest path to be congested while links on alternative paths are under-utilised.

This property of conventional IP routing protocols, of traffic aggregation on the shortest path, can cause sub-optimal use of network resources, and can consequently impact the SLAs that can be offered (or require more network capacity than is optimally required).

MPLS traffic engineering (TE) [2] uses the implicit MPLS characteristic of separation between the data plane (also known as the forwarding plane) and control plane to allow routing decisions to be made on criteria other than the destination IP address in the IP header, such as available link bandwidth. MPLS TE effectively provides an explicit routing capability at Layer 3, allowing paths to be used other than the shortest cost path to a destination, thereby avoiding traffic aggregation on the shortest path and providing more optimal use of available bandwidth.

MPLS TE uses the following mechanisms:

- Information on available network resources, including a pool of available bandwidth maintained per link, are flooded by means of extensions to link-state based IP routing protocols such as IS–IS [26] and OSPF [17].
- A constraint-based routing (CBR) algorithm is used to compute the traffic path based upon a fit between the available network resources (advertised via IS–IS or OSPF) and the resources required, i.e. a requested amount of bandwidth.
- The Resource ReSerVation Protocol (RSVP) [5], with enhancements for MPLS TE [1], is used to signal and maintain an explicit route (termed a ''traffic engineered tunnel''), from *head-end* to *tail-end*, in the form of an MPLS Label Switched Path (LSP). This LSP follows the path determined by the constrain-based routing algorithm. In signalling the tunnel, admission control is performed at every hop.
- Traffic routed onto these LSPs or tunnels will then follow the traffic engineered explicit route to the destination, rather than the conventional IGP shortest path.

The following conditions can all be drivers for the deployment of MPLS traffic engineering [28]:

- *Network asymmetry.* Asymmetrical network topologies can often lead to traffic being aggregated on the shortest path whilst other viable paths are under-utilised. Network designers will often try to ensure that networks are symmetrical such that where parallel paths exist, they are of equal cost and hence the load can be balanced across them using conventional IGPs. Ensuring network symmetry, however, is not always possible due to economic or topological constraints. Traffic engineering offers obvious benefits in these cases.
- *Unexpected demand.* In the presence of unexpected traffic demand (e.g. due to some new popular content), there may not be enough capacity on the shortest path (or paths) to satisfy the demand. There may be capacity available on non-shortest paths, however, and hence traffic engineering can provide benefit.
- *Long bandwidth lead-times.* There may be instances when new traffic demands are expected and new capacity is required to satisfy the demand, but is not available in suitable timescales. In these cases, traffic engineering can be used to make use of available bandwidth on non-shortest path links.

The use of TE gives the service provider flexibility in how to manage their backbone bandwidth in order to achieve their SLAs. The more effective use of bandwidth potentially allows higher service availability targets to be offered with the existing backbone bandwidth. Alternatively, it offers the potential of achieving the existing service availability targets with less backbone bandwidth.

5.2. Diffserv and TE

MPLS TE and Diffserv can be deployed concurrently in an IP backbone, with TE determining the path that traffic takes on aggregate based upon aggregate bandwidth constraints, and Diffserv being used on each link for differential scheduling of packets on a per class basis. Whilst TE and Diffserv are orthogonal technologies they can be used in concert for combined benefit: TE allows distribution of traffic on non-shortest paths for more efficient use of available bandwidth, whilst Diffserv allows over/under-provisioning ratios to be determined on a per class basis.

MPLS TE, however, computes tunnel paths for aggregates across all traffic classes and traffic from different classes may use the same TE tunnels. MPLS TE is aware of only a single aggregate *global pool* of available bandwidth per link and is unaware of what specific link bandwidth resources are allocated to which queues, and hence to which class.

Consequently, MPLS TE has been extended with Diffserv-aware traffic engineering (DS-TE) [10], which introduces the concept of an additional and more restrictive pool of available bandwidth on every link. This more restrictive bandwidth pool is termed the *sub-pool*, while the regular TE bandwidth is called the *global pool* (the sub-pool is a portion of the global pool). The sub-pool may be used for constraint-based routing and admission control of tunnels for "guaranteed" or EF class traffic and the global pool used for regular (non-"guaranteed") or AF class traffic.

In supporting DS-TE, extensions have been added to IS–IS and OSPF [11] to advertise the available sub-pool bandwidth per link as well as the available global-pool bandwidth. In addition, the TE constraint-based routing algorithms have been enhanced for DS-TE in order to take into account the constraint of available sub-pool bandwidth in computing the path of sub-pool tunnels. RSVP has also been extended [11] to indicate if it is signalling a sub-pool or global-pool tunnel.

It is understood that setting an upper bound on the EF class (e.g. VoIP) effective utilization per link allows a way to restrict the effects of delay and jitter due to accumulated burst [4,7]. DS-TE can be used to assure that this upper bound is not exceeded. For example, consider the network in Fig. 6 DS-TE could be used to ensure that traffic is routed over the network so that, on every link, there is never more than 25% (or any assigned percentage) of the link capacity for EF class traffic, whilst there can be up to 100% of the link capacity for EF and AF class traffic in total. Each link in Fig. 6 is 100 Mbps. R1 is sending an aggregate of 40 Mbps of traffic to R8, and R2 is also sending an aggregate of 40 Mbps of traffic to R8. An IGP and TE metric value of one is applied to each link.

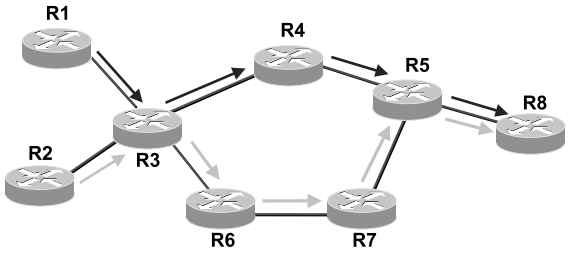

Fig. 6. DS-TE deployment example.

In this case, both the IGP and non-Diffserv aware TE would pick the same route. The IGP would pick the top route (R1/R2 → R3 → R4 → R5 → R8) because it is the shortest path (metric of 4), whilst TE would pick the same path because it is the shortest path that has sufficient bandwidth available (metric of 4, 100 Mbps bandwidth available, 80 Mbps required). The decision to route both traffic aggregates via the top path may not seem appropriate if we examine the composition of the aggregate traffic flows.

If each of the aggregate flows is comprised of 5 Mbps of VoIP traffic and 35 Mbps of business data traffic, then in this case the total VoIP traffic load on the top links will be 10 Mbps, which is within our EF class bound of 25%. If, however, each traffic aggregate is comprised of 20 Mbps of VoIP and 20 Mbps of business data traffic then such routing would aggregate 40 Mbps of VoIP traffic on the R3 → R4 → R5 links, thereby exceeding our EF class bound of 25%. DT-TE can be used to overcome this problem: each link is configured with an available global pool bandwidth of 100 Mbps, and an available sub-pool bandwidth

of 25 Mbps (i.e. 25% of 100 Mbps). A global-pool tunnel of 20 Mbps is then configured from R1 to R8 for business data traffic, and a sub-pool tunnel of 20 Mbps for VoIP traffic. Similarly, from R2 to R8 a global-pool tunnel of 20 Mbps is configured for business data traffic, and a sub-pool tunnel of 20 Mbps for VoIP class traffic. The DS-TE constraint based routing algorithm would then route the sub-pool tunnels to ensure that the 25 Mbps bound is not exceeded on any link, and of the tunnels from R1 and R2 to R8, one sub-pool tunnel would be routed via the top path (R1/R2 → R3 → R4 → R5 → R8) and the other via the bottom path (R1/R2 → R6 → R7 → R5 → R8). [13] In this particular case, there would be enough available bandwidth for both global pool tunnels to be routed via the top path (R1/R2 → R3 → R4 → R5 → R8), which has the shortest metric.

DS-TE enables service providers to perform separate route computation and admission control for different classes of traffic. This enables the distribution of EF and AF class load over all available EF and AF class capacity making optimal use of available capacity. It also provides a tool for constraining the EF class utilization per link to a specified maximum thus providing a mechanism to help bound the delay and jitter. In order to provide these benefits, however, the configured bandwidth for the sub-pool and global pool must represent queuing resources, which are only available for traffic-engineered traffic, and hence non-traffic engineered traffic should be queued separately on each link. By combining DS-TE with Diffserv queuing mechanisms on each link, the service provider can offer tight-SLAs for EF class traffic (such as VoIP) with admission control without large over-provisioning of capacity.

6. MPLS traffic engineering fast reroute

In Section 4, we highlighted that link or node failures in an IP backbone can significantly impact the availability that can be offered for SLAs across all classes. Whilst sub-second convergence for IP routing protocols is a realistic prospect, it is expected that IGP convergence will not be able to match the capabilities of SDH/SONET networks, which use the capabilities of Multiplexer Section Protection (MSP) and Automatic Protection Switching (APS) respectively to recover around failures in tens of milliseconds. This is because the functions are performed in fundamentally different ways: IGP convergence is based on a distributed computation [14] whereas SDH/SONET restoration is based upon local detection and pre-computed local protection around the failure.

MPLS traffic engineering fast reroute (FRR) extends the concepts of local failure detection and protection to MPLS TE in order to provide very rapid recovery around failures (e.g. a few tens of milliseconds) prior to any distributed convergence/reoptimisation. Without FRR, under failure conditions, the head-end of a TE tunnel determines a new route for the tunnel LSP. Recovery at the head-end provides for the optimal use of resources, however, due to messaging delays, the head-end cannot recover as fast as is possible by making a repair at the point of failure.

MPLS TE Fast Reroute adds additional capabilities to MPLS TE in that it provides local protection of tunnel LSPs in the presence of link failure. This enables all traffic carried by tunnel LSPs that traverse a failed link to be rerouted around the failure. The reroute decision is completely controlled locally by the router interfacing the failed link.

MPLS TE FRR uses the following mechanisms:

- *Hierarchical LSPs.* FRR uses the concept of hierarchical LSPs; the protected tunnel LSPs are switched rapidly into backup tunnel LSP at the point of failure (also known as the point of local repair). The backup tunnel-LSP provides an explicit route around the failed link or node. To support FRR, extensions have been added to

[13] A propagation-delay constraint can also be specified for the sub-pool tunnels to ensure that the chosen path exhibits a propagation delay smaller or equal to the specified value [12].

[14] The local node next to the failure distributes the information and then it is up to all the other nodes to each compute their routing table to route around the failure.

RSVP [27] to indicate which tunnels are fast reroutable and to carry additional label information required for FRR. For maximum protection, the path for the backup tunnel LSP can be determined to use disjoint resources (such as optical channels, fibres or ducts) from the protected node or link.

- *Rapid failure detection.* If POS links are used, SDH/SONET signalling allows link or node failures to be detected by directly connected nodes in less than 10 ms. Where POS links are not used, RSVP hellos can be used between adjacent nodes for failure detection.
- *Rapid local protection.* Rapid local protection around the failure is possible because the switching entries for the backup tunnel-LSP are precomputed. Once a failure is detected, the local node has only to copy the pre-computed backup tunnel-LSP switching entry into its switching table for recovery around the failure to be achieved.
- *Path reoptimisation.* The head-end of the tunnel is also notified of the link failure through the IGP or through RSVP; the head-end may then attempt to establish a new, and possibly more optimal, LSP that bypasses the failure, using make-before-break signalling with RSVP. This is a significant advantage compared to SDH/SONET protection.

The local nature of FRR allows very rapid protection and restoration around failures in IP backbone networks. For SDH/ SONET links, detecting the failure of a link is typically done in less than 10 ms, and with FRR, many hundreds of protected tunnel-LSPs can be switched around the failure in less than 50 ms. This is equivalent to the level of protection provided by MSP and APS and in SDH and SONET networks respectively.

FRR is designed for backbone deployment where the number of network components is typically relatively low, but where the failure of those components can have severe impacts on services and SLAs. The determination of optimal routing for FRR backup tunnels in different failure scenarios is, however, a complex problem and needs to take into account factors including the available bandwidth on potential backup paths, tunnel inter relationships and interdependencies on the lower layer network topologies. This subject is currently the focus of further research and development efforts.

For the service provider, FRR provides the capability to significantly improve the perception of service quality for IP telephony users. For IP telephony, if IP connectivity is lost for a few hundreds of milliseconds, the users will perceive a glitch in their call. By deploying FRR to protect key network resources, service providers can ensure that link failures are imperceptible to IP telephony users and can hence offer the highest availability of service for VoIP class traffic. Further, enabling this protection technique at the IP layer allows for better statistical multiplexing, better reoptimisation upon active protection and cost advantage due to the consolidated architecture, when compared to providing this capability at lower layers, using SDH/SONET capabilities for example.

7. Conclusion

In the context of ever more competitive service provider offerings and ever more demanding requirements from their customers, this paper has analysed which SLA parameters are significant for IP service performance (delay, jitter, loss bandwidth/throughput, per-flow sequence preservation and availability) and has listed the targets usually set for these parameters for the typical backbone aggregated classes of service.

We have reviewed the technologies that can be used to tighten these SLAs and hence serve as foundation for a multiservice IP backbone: Diffserv, fast IGP convergence, traffic engineering, Diffserv-aware traffic engineering and MPLS TE FRR.

The Diffserv analysis showed that there should be an important economical benefit in leveraging this technology, which enables under- or over-booking to be performed on a per class rather than aggregate basis. This per class under-/over-booking capability enables tight-SLAs to be offered for some classes of traffic without aggregate over-provisioning of capacity, leading to savings in

terms of required bandwidth. It was noted that the typical Diffserv deployment strategy targeted by current designs is relatively simple and only requires a one-time router configuration. The results of router-based testing finally illustrates the tight-SLA capabilities of today's high-performance routers and confirmed that the previously defined SLA targets could easily be met. For these reasons, a multiservice IP backbone design should leverage Diffserv technology.

Recent developments in IGP implementations have resulted in significant improvements in IP backbone IGP convergence with convergence times 1–2 s being pragmatically achievable today. This reduction in convergence times allows higher availability targets to be offered for SLAs across all service classes. Consequently, fast IGP convergence is also recommended as a foundation of multiservice IP backbone network designs

MPLS traffic engineering gives the service provider the capability to use available backbone bandwidth more effectively. Either it allows higher service availability targets to be offered with the existing backbone bandwidth or, alternatively, it offers the potential of achieving the existing service availability targets with less backbone bandwidth. Notwithstanding these benefits, it is noted that not all networks will benefit from the deployment of TE

Diffserv-aware MPLS traffic engineering extends the base capabilities of TE to allow route computation and admission control to be performed separately for different classes of service. By combining DS-TE with Diffserv mechanisms on each link, the service provider can offer tight-SLAs for EF class traffic (such as VoIP) with admission control without large over-provisioning of capacity.

Finally, for networks seeking sub-100 ms convergence, MPLS TE fast reroute provides the capability for protection around failures at the IP Layer. By deploying FRR to protect key network resources, service providers can ensure that link failures are imperceptible to IP telephony users and can hence offer the highest availability of service for VoIP class traffic.

In summary, the foundation for a multiservice tight-SLA IP network consists in a Diffserv and Fast-IGP design. This design is complemented by MPLS technologies such as traffic engineering, Diffserv-aware traffic engineering and fast reroute depending on the specific context and requirements of the considered network.

Acknowledgements

The authors would like to thank Olivier Bonaventure, Thomas Telkamp and Bruce Thompson for their valuable feedback and constructive comments.

References

[1] D.O. Awduche et al., RSVP-TE: extensions to RSVP for LSP tunnels, RFC3209, December 2001.
[2] D. Awduche et al., Requirements for traffic engineering over MPLS, RFC 2702, September 1999.
[3] S. Blake et al., RFC2475, An Architecture for Differentiated Service, December 1998.
[4] T. Bonald, A. Proutiere, J. Roberts, Statistical guarantees for streaming flows using expedited forwarding, INFOCOM 2001.
[5] R. Braden et al., RFC2205, Resource Reservation Protocol (RSVP) Version 1 Functional Specification, September 1997.
[6] S. Casner, C. Alaettinoglu, C.-C. Kuan, A fine-grained view of high-performance networking, Packet Design, NANOG 22, May 20–22, 2001.
[7] A. Charny, J.-Y. Le Boudec, Delay bounds in a network with aggregate scheduling, in: First International Workshop on Quality of Future Internet Services, Berlin, Germany, 2000.
[8] B. Davie (Ed.) et al., An expedited forwarding PHB, Internet draft, April 2001 Charny et al., An expedited forwarding PHB, Internet Draft.
[9] N.G. Duffield et al., A flexible model for resource management in virtual private networks, SIGCOMM'99.
[10] Le Faucheur et al., Requirements for support of Diff-Serv-aware MPLS Traffic Engineering, Internet Draft.
[11] Le Faucheur et al., Protocol extensions for support of Diff-Serv-aware MPLS Traffic Engineering, Internet Draft.
[12] Le Faucheur et al., Use of IGP Metric as a second TE Metric, Internet Draft.
[13] S. Floyd, V. Jacobson, Random early detection gateways for congestion avoidance, IEEE/ACM Transactions on Networking 1 (4) (1993) 397–413.
[14] J. Heinanen et al., RFC2597, Assured Forwarding PHB Group, June 1999.
[15] http://www.ietf.org/html.charters/ippm-charter.html.

[16] V. Jacobson et al., RFC2598, An expedited forwarding PHB, June 1999.
[17] D. Katz et al., Traffic engineering extensions to OSPF, Internet Draft.
[18] M. Laor, L. Gendel, Effect of packet reordering in a backbone link on applications throughput, submitted to IEEE Network Magazine.
[19] M. Mathis, The macroscopic behavior of the TCP Congestion Avoidance Algorithm, Computer Communication Review, July 1997.
[20] S. McCreary, K. Claffy, Trends in wide area IP traffic patterns—a view from Ames Internet Exchange, in: Proceedings of 13th ITC Specialist Seminar on Internet Traffic Measurement and Modelling, Monterey, CA, 18–20 September, 2000.
[21] K. Nichols et al., RFC2474, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, December 1998.
[22] http://www.ietf.org/html.charters/pwe3-charter.html.
[23] J.M. McQuillan, I. Richer, E.C. Rosen BBN Report 3803, ARPANET Routing Algorithm Improvements, First Semiannual Technical Report, April 1978.
[24] J.M. McQuillan, I. Richer, E.C. Rosen, The New Routing Algorithm for the ARPANET, IEEE Transactions on Communications, May 1980.
[25] M. Shreedhar, G. Varghese, Efficient fair queuing using deficit round robin, SIGCOMM 1995.
[26] H. Smit, T. Li, ISIS extensions for traffic engineering, Internet Draft.
[27] G. Swallow, R. Goguen, RSVP Label allocation for backup tunnels, Internet Draft.
[28] T. Telkamp, Hot Interconnect, Standford 2001.

Clarence Filsfils has an Engineering Degree in Computer Sciences from the Institute Montefiore of the University of Liege, Belgium and a Business Degree from the Solvay Business School, Brussels Belgium. He joined Cisco in 1996 and as a Distinguished Engineer; he focuses on IP Core Routing and Capacity Management (IP QoS/Traffic Engineering) designs.

John Evans received a B.Eng. (Hons) degree in Electronic Engineering from the University of Manchester Institute of Science and Technology (UMIST), UK in 1991 and an M.Sc. in Communications Engineering from UMIST in 1996. He joined Cisco Systems in 1998 and as a Consulting Engineer; he focuses on IP network design and development with special interests in core routing and traffic management, including IP quality of service and traffic engineering. Prior to joining Cisco, he worked on the design and development of large-scale networks for the financial community and for the military.