A Service-Centric IP Quality of Service Architecture for Next Generation Networks

D. Goderis, S. Van den Bosch, Y. T'Joens
Alcatel Network Strategy Group
Fr. Wellesplein 1, 2018, Antwerpen, Belgium
{Danny.Goderis, sven.van_den_bosch, Yves.Tjoens}@alcatel.be

G. Pavlou, P. Trimintzios
Centre for Communication Systems Research, University of Surrey
Guildford, Surrey GU2 7XH, U.K.
{G.Pavlou, P.Trimintzios}@eim.surrey.ac.uk

P. Georgatsos
Algonet S.A.
Syngrou Av., 206, 176 72, Athens, Greece
pgeorgat@egreta.gr

G. Memenios, E. Mykoniati
National Technical University of Athens
Heroon Polytechniou 9, 157 73 Zografou, Athens, Greece
{gmemen, mykel}@telecom.ntua.gr

D. Griffin
Dept of Electronic and Electrical Eng.
University College London
Torrington Place, London WC1E 7JE, U.K. D.Griffin@ee.ucl.ac.uk

C. Jacquenet
France Telecom R&D
Rue des Coutures 42, BP6243, 14066 Caen Cedex 04, France
Christian.Jacquenet@francetelecom.fr

Abstract

IP Differentiated Services is widely seen as the framework to provide Quality of Service (QoS) in the Internet in a scalable fashion. However many issues have still not been fully addressed, such as: the way Per-Hop Behaviours can be combined to provide end-to-end services; the specification of admission control and resource reservation mechanisms; and the role of management plane functionality and its integration with the control and data planes. This paper presents the Service Management aspects of an integrated control and management architecture for supporting end-to-end QoS-based IP services in Next Generation Networks. It introduces a two-phased approach for service negotiation, namely, service subscription followed by service invocation, and describes the interworking between service and resource management based on the concept of a *resource provisioning cycle*.

Keywords

IP Services, Quality of Service, Differentiated Services, Service Level Specifications, Service Management, Resource Provisioning

0-7803-7382-0/02/$17.00 ©2002 IEEE

1 Introduction

Today the Internet attempts to deliver traffic as soon as possible within the limits of its abilities, but without any guarantees related to throughput, delay, inter-packet delay variation (jitter) and packet loss. So far this so-called best-effort forwarding paradigm has worked well because most IP applications are low-priority and low-bandwidth data applications with high tolerance on delay and delay-variation. Value-added IP services, however, like Voice over IP (VoIP) and other multimedia applications, require stringent end-to-end Quality of Service (QoS) guarantees. Therefore, the key challenge for Next Generation Networks (NGNs) is to extend IP-based networks with scalable, multi-service QoS capabilities, while still providing the key advantages of IP that made the Internet possible. On these multi-service networks, operators will have to honour complex Service Level Agreements (SLAs), acknowledging different types of traffic in terms of bandwidth requirements, delay and other QoS parameters.

Within the Internet Engineering Task Force (IETF) several IP QoS technologies have been proposed. *Integrated Services* (IntServ) was the first proposal [1], based on per-flow resource reservation and admission control through the Resource reSerVation Protocol (RSVP) [2]. The main disadvantage is that the required *per-flow* state information and QoS treatment in the core IP network pose severe scalability problems. These problems led to the development of the *Differentiated Services* architecture [3], which allows for flow aggregation in order to deal with the scalability issues.

Differentiated Services (DiffServ) is based on the marking of IP packets with priority information, the so-called Differentiated Services Code Point (DSCP), which is a 6-bit encoded field of the Differentiated Services (DS) byte of an IP header [4]. DiffServ capable routers implement different packet forwarding behaviours, called Per Hop Behaviours (PHB), for distinct traffic types based on the DSCP-value in the IP packet header. This differential treatment of aggregate packet streams, i.e. on a per DSCP basis, makes DiffServ routers scalable, even at Gigabit link rates. The DiffServ technology maintains scalability in the core routers by pushing major complexity to the edges of the network and also to the management plane.

DiffServ is clearly a promising technology; however to deliver real-time multimedia services on DiffServ-based IP networks still requires a substantial amount of further research and development:

- The DiffServ architecture offers the network operator a number of elementary QoS building blocks, including the PHBs and the Traffic Conditioning Block (TCB). The way PHBs should be concatenated to emulate Virtual Leased Lines (VLLs), for example, is not part of DiffServ as it has been developed to-date.
- The IETF recently defined the notion of DiffServ *edge-to-edge* packet behaviours, i.e. Per Domain Behaviours (PDBs) [5], including the Virtual Wire PDB and Assured Rate PDB. However, the concept of service classes and the definition of IP transport services remains vague.

- DiffServ provides service differentiation for aggregate IP packet streams by implementing different PHBs for different DSCP values. However it is unclear how QoS guarantees can be committed to e.g. individual multimedia services such as voice and video streams. Especially the required trade-off between scalability and per-multimedia-flow resource reservation and admission control is an open research issue.
- Network management plays a key role in the provisioning of value-added IP services over DiffServ networks. Every router must be configured so that sufficient resources in terms of bandwidth and buffer space are available to support the SLAs that have been contractually agreed between a customer and a service provider. This type of functionality has been described in [6] as a Bandwidth Broker.

TEQUILA (Traffic Engineering for Quality of service in the Internet at Large) [7] is a European project looking at these outstanding research aspects of the DiffServ architecture. Its primary goal is to provide an integrated architecture and associated techniques for end-to-end QoS delivery in a DiffServ-capable IP network [8]. The project deals with both service and resource management aspects and the relationships between them. In addition, both Multi-Protocol Label Switching [9] and IP-based techniques for traffic engineering are investigated [10].

This paper focuses on the Service Management aspects of provisioning QoS-based IP services as developed by the TEQUILA project and is structured as follows. In Section 2 a layered service model for DiffServ is presented, explaining the concepts of IP transport services and QoS classes. Section 3 is the main section of the paper and outlines our proposed model for structuring the functionality required by a system for the dynamic provisioning of the resources participating in the delivery of end-to-end QoS. Section 4 illustrates the ideas presented through two scenarios, a corporate IP VPN and a NGN architecture where an IP backbone connects a number of Trunking Gateways. Section 5 outlines a suitable protocol for negotiating services and their QoS capabilities. The final section presents our summary and conclusions.

2 A Layered Service Model for IP Differentiated Services

One of the basic DiffServ QoS concepts is the PHB, exposing, in a generic way, the QoS capabilities of a router. PHBs may be implemented by a range of scheduling and buffering mechanisms such as Priority Queuing, Weighted Fair Queuing and algorithms for implementing packet dropping policies such as Random Early Detection. The PHB is the basic building block for supporting value-added IP services, previously negotiated between the provider and its customers through SLAs. However, there is a missing link between the low-level data-plane concept of a PHB and a high-level IP transport service such as VoIP. This is illustrated in Figure 1. The upper two layers of the figure are discussed in section 2.1 while the lower layers are discussed in section 2.2.

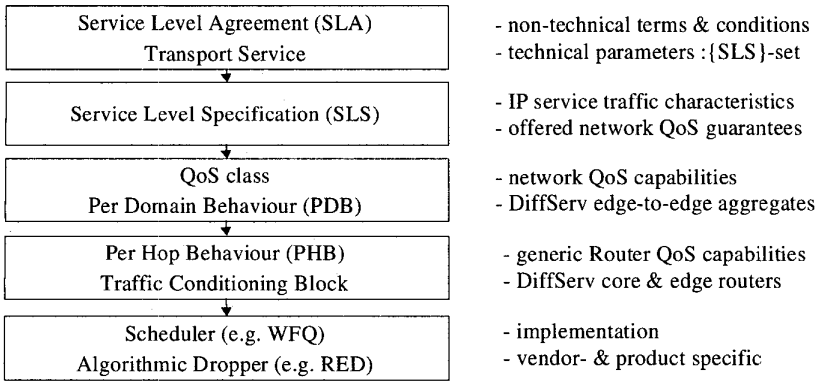

Figure 1: A DiffServ Layered Service Model

2.1 Service Level Specifications

The upper two layers of Figure 1 describe the interface between the IP transport provider and the customer. According to the IETF DiffServ working group, a Service Level Agreement (SLA) is *"the documented result of a negotiation between a customer and a provider of an IP service that specifies the levels of* availability, *serviceability, performance, operation or other attributes of the transport service"* [11]. The SLA contains technical and non-technical terms and conditions. The technical specification of the IP connectivity service is given in Service Level Specifications (SLSs). A SLS *"is a set of technical parameters and their values, which together define the IP service, offered to a traffic stream by a DiffServ domain"*. SLSs describe the traffic characteristics of IP flows and the QoS guarantees offered by the network to these flows. Note that a SLA may contain a set of SLSs. Our definition of a SLS [12] is uni-directional, thus requiring two symmetric SLSs to describe services such as a bi-directional Virtual Leased Line (VLL) or a telephone call.

The DiffServ working group does not intend to specify further the content of a SLS beyond the loose definitions given above. Nevertheless, the definition of a SLS is a key-step towards the provisioning of value-added IP services because it specifies the semantics of the interface between the provider and the customer, i.e. *the technical terms and conditions.* To this end, we have proposed a standard template for the parameters and semantics of a SLS [12]. The basic parameter groups of the SLS template with a brief description are presented in Table 1.

Parameter Group	*Description*
Customer Identifier	Identifies the customer or the user for Authentication, Authorisation and Accounting purposes (AAA)
Flow Descriptor	Identifies *the packet stream* of the contract by e.g. specifying a packet filter (DSCP, IP source address, etc).
Service Scope	Identifies the administrative region *where* the contract is applicable by e.g. specifying ingress and egress interfaces.
Service Schedule	Specifies *when* the contract is applicable by giving e.g. operating hours of the service on a per-day, per-month, etc. basis
Traffic Descriptor	Describes the traffic envelope through e.g. a token bucket algorithm parameters, allowing to identify in- and out-of-profile packets
QoS Parameters	Specifies the QoS network guarantees offered by the network to the customer for in-profile packets including delay, inter-packet delay variation, packet loss and throughput guarantees.
Excess Treatment	Specifies the treatment of the out-of-profile packets at the network ingress edge including dropping, shaping and re-marking.

Table 1: SLS Parameters

2.2 Network QoS Layer

The third layer in Figure 1 is the "network QoS layer" mediating between the customer-specific SLS-based services and the elementary PHBs supported by the routers. The notion of the QoS-class is introduced to substantiate this mediation. QoS classes expose the network-wide QoS transport capabilities and they are bound to the specific technology employed and capabilities provided by the network. For example, a Virtual Wire (VW) QoS-class could be defined to denote an edge-to-edge transport capability with a guaranteed maximum packet delay and a guaranteed throughput for an aggregate IP packet stream marked as Expedited Forwarding (EF). QoS-classes should be seen as specifications of a Per Domain Behaviour [5]. We have adopted the following specification of a QoS class.

Parameter	*Description*
Ordered Aggregate	The allowed values are: Expedited Forwarding (EF), Assured Forwarding 1-4 (AF1, AF2, AF3, AF4), Best Effort (BE)
Delay	The *delay* is the maximum *edge-to-edge* delay that the in-profile packets of a certain IP stream should experience. It is a continuous parameter that may be worst case (deterministic) or percentile (probabilistic).
Packet Loss	The *packet loss* is the upper bound of the *edge-to-edge* packet loss probability that in-profile profile packets of an IP stream should have.

Table 2: Specification of a DiffServ QoS Class

A finite number of QoS-Classes is obtained by allowing only a discrete number of possible delay and loss values. The delay-loss ranges are mainly driven by the corresponding performance parameters of the services offered (expressed in the SLSs) and they are subject to the capabilities and characteristics of the network including its topology. Furthermore, they may be policy-influenced, changing from time to time as service and network policies warrant so.

A network supports certain QoS classes through deploying dedicated Traffic Conditioning Block (TCB) at the edge routers, PHBs throughout the network, and an overall resource management system (Section 3). Supporting customer specific SLSs boils down to a "service mapping" of the SLS to a QoS class and SLS admission control, while the network should be suitably engineered to gracefully sustain the traffic of the admitted SLSs. The service related aspects (mapping SLSs to QoS-classes and SLS admission) is the focus of the paper and will be explained in the following section.

3 IP Service and Resource Management

A key aspect in the offering of value-added IP services is service-driven QoS-based resource management. Providers and customers negotiate agreements through SLAs and the provider must then assure that sufficient resources are provisioned to support these agreements. While the need for Bandwidth Broker (BB) capabilities has been identified [6], its architecture and related admission and reservation aspects remain largely unspecified. Therefore we substantiate and extend the notion of the BB by defining the functionality of an integrated management and control system that combines both service negotiation/invocation and traffic engineering aspects. Our functional architecture has been proposed in [8] and the main components are depicted at a high level in Figure 2. The architecture can be realised in several ways: as a single centralised entity or, for scalability, as a more sophisticated hierarchical system, logically and physically distributed.

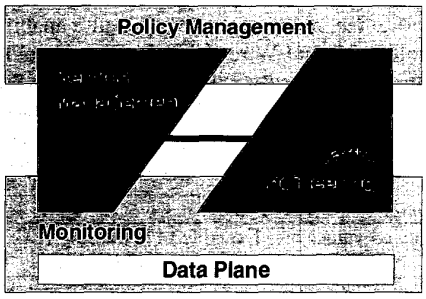

Figure 2: Overall Functional Model for Service and Resource Management of QoS-based IP networks

The "low-level" data plane includes the DiffServ PHB and TCBs, while the "high-level" policy management has a supporting role and allows administrators to define and enforce policies on the Service Management and TE subsystems in an automated fashion. This means that new policies specified in a high-level declarative manner are introduced in the system and evaluated dynamically, without the need for system re-engineering. Monitoring is a supporting sub-system that is not further detailed in this paper.

The Service Management (SM) and Traffic Engineering (TE) sub-systems are the major essential parts of the TEQUILA system architecture. SM includes service creation, negotiation and assurance. Service creation is the process of defining services and service classes by the provider. The SLS template proposed in [12] can be used to this end. Service negotiation is the process for subscribing to and subsequently invoking value-added services between provider and customer. We consider this operational "on-line" process as critical to the overall solution of providing QoS-based services and this constitutes the main subject of this paper.

Service assurance enables the operator to verify whether QoS performance guarantees as committed in SLAs are in fact being met. This requires an in-service verification of throughput, delay and packet loss characteristics. Service assurance operates on the statistical data gathered by network monitoring through the network elements.

TE is the process of specifying the manner in which traffic is treated within the network. TE has both customer and system-oriented objectives. The customers expect certain performance from the network, depending on the type of traffic specified in their SLSs. The provider, on the other hand, attempts to satisfy customer requirements in a cost-effective manner. Hence, the target is to accommodate as many traffic requests (SLSs) as possible by optimally using the available network resources. In this paper we concentrate on SM and its interactions with TE but our approach to service-driven resource management is described in more detail in [10].

Our SM system is further decomposed into functional blocks as shown in Figure 3. The figure only shows those functions involved in "operational" SM. Service creation and service assurance are not considered further in this paper.

The SM system of the provider contains three main functional components: service subscription, service invocation and traffic forecast. The customer interacts with the provider's system through service subscription and invocation. Traffic forecast is the glue providing the main interface between the SM and TE systems.

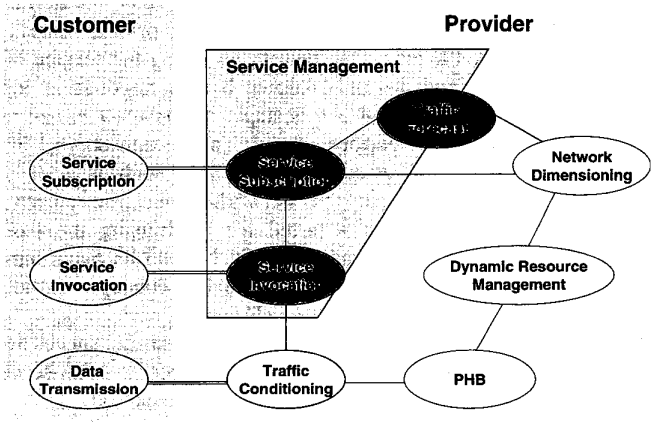

Figure 3: Service Management Functions and Interactions

3.1 Service Subscription

Service Subscription refers to the period during which an IP transport service is requested by the customer, negotiated with the service provider and agreed upon by both parties. Successful negotiation results in a SLA containing, among other aspects, the technical description of the IP transport service, which are based on the SLS template. The customer, as a legal entity, may be a peering Internet Service Provider (ISP), an Application Service Provider (ASP), an organisation or an individual residential user. For example the SLA could specify a VLL connecting two sites of a company or an IP VPN connecting two Trunking gateways owned by a VoIP service provider. These examples are elaborated further in section 4.

Service subscription allows the network provider to plan, dimension and traffic engineer its network on the basis of the traffic implied by the subscriptions. It assures the customer regarding (future) resource availability for the traffic envelope specified in the contract. The following figure further decomposes this process.

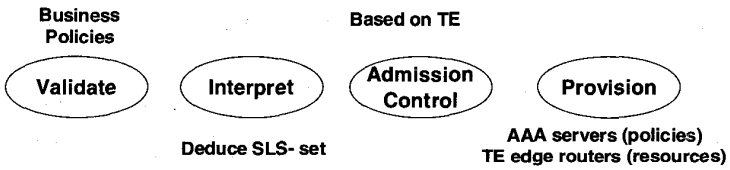

Figure 4: Decomposition of the Service Subscription process

The *validation* process is the "admission control" process with regard to business policy and administrative information. In this context acceptance of new subscriptions, e.g. for IP VPNs, typically depends on customer profiles and other business agreements. The *interpret* process depends on the way the IP transport service is specified in the SLA. If all the technical parameters of the IP service are explicitly specified by the full {SLS}-set values then no interpretation is required. However, a provider may also offer a "gold leased line" or "fast Internet access", for example, in which case the technical details describing the service offering are hidden from the customer and are only known by the negotiation logic of the provider and they must be interpreted at this stage. *Admission Control* realises the subscription negotiation logic. It performs "static" admission control in the sense that it determines whether a requested long-term SLS, such as a VLL or an IP VPN, can be supported or not in the network given the current network configuration. Admission Control is required for minimising the likelihood of overwhelming the network with customer contracts beyond its maximum capacity, as defined by the TE subsystem. It is based on the concept of the *Resource Provisioning Cycle*, which is explained further (Figure 5).

If the validate, interpret and admission control processes are successful then the customer becomes a subscriber and the provider configures its AAA servers and its edge routers with the appropriate traffic conditioning information.

3.2 Service Invocation

Service Invocation refers to the epoch during which users, or their applications, request resources and, if successful, traffic is injected into the network. Users may be employees of the organisation having subscribed to leasing a VLL. They may also be residential customers of ASPs offering voice services by connecting trunking gateways over a data network, for example.

Service Invocation may be an *implicit* or an *explicit* process. In the former case, no actual invocation is required and the users may directly inject packets into the network based on their SLA contract (agreed during service subscription). The SLA can be e.g. a corporate IP VPN describing connectivity information and overall throughput guarantees between sites. There may be no need for per-application or per-call (flow) awareness at the edge routers of the ISP, depending on the type of IP VPN technology deployed. The edge routers are only aware of the aggregate SLA subscription contract. Service differentiation within the SLA-contract is still possible based on e.g. the DSCP value of the packets.

In the context of IP NGNs, *service invocation* is an explicit process related with the *per-multimedia call* admission control and resource reservation. The process can be decomposed in a similar fashion to Figure 4. There are however two main differences between the subscription and invocation processes.

- *Validation* consists in checking whether authentication and authorisation conforms to the information already provided by subscription, e.g. by checking whether the user is authorised to invoke that service.
- *Admission control* checks whether the request related to the multimedia call (e.g. 64 Kbps) still fits into the overall throughput guarantee offered by the subscription contract (e.g. an "E1" virtual leased line of 2 Mbps), and furthermore ensures that there is sufficient capacity in the network to admit the requested traffic.

Finally, a service invocation request may be negotiated *in-band* or *out-of-band*. In-band negotiation takes place directly over the router ingress interface towards the provider's network, based on, for example, the RSVP protocol. Out-of-band negotiation may be realised by a dedicated multimedia call signalling protocol (see Figure 7).

3.3 Traffic Forecast

Traffic Forecast (TF) generates a traffic estimation matrix (TM) based on a repository of {SLS}-subscriptions. The TM specifies the anticipated traffic demand per ingress-egress pair and per QoS-class:

TM = {QoS-class | ingress-egress | min-demand - max-demand};
QoS-class = {OA | max delay | max packet loss}

Being of statistical nature, the anticipated traffic demand is specified in terms of a range (from a minimum to a maximum). The *maximum demand* is calculated such that if the network could provide this capacity then the QoS guarantees specified in all SLSs would *always* be fulfilled. This value is obtained by summing SLS-throughput guarantees, without any time-variant statistical multiplexing gain. The

minimum demand takes into account possible over-provisioning policy rules, monitoring information, the physical nature and capacity of the access links, etc. The value is such that, under "reasonable" operational conditions, the QoS guarantees of the SLSs are "almost" always fulfilled. The definitions of reasonable and almost are left as configurable parameters which may be modified by the policy system according to the business objectives of the network operator.

Figure 5 shows that the calculation of the TM involves three basic actions. A service-mapping algorithm translates the QoS requirements defined in the SLSs into a (predictive) form that complies with the TM specification. An aggregation algorithm combines entries from different SLSs with the same ingress-egress context and QoS-class into a single entry. At this stage over-provisioning rules can be taken into account for calculating minimum demand. Finally an forecast algorithm can be specified taking also into account traffic projections and historical data.

Figure 5 also illustrates that Traffic Forecast is the "glue" between the customer-oriented SM sub-system and the resource-oriented TE sub-system. The input of TF is *SLS (customer) aware* while the output is only *QoS-Class aware.*

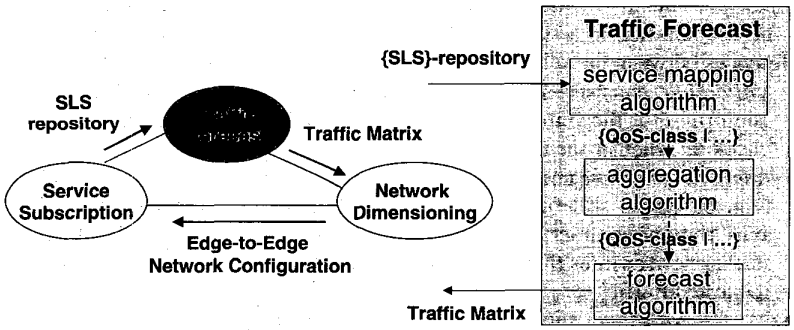

Figure 5: Traffic Forecast and the Resource Provisioning Cycle

It should be noted that for scalability reasons the TE subsystem should *by no means* be SLS-aware. On the other hand, and also for scalability reasons, the SM subsystem should have no knowledge about internal network configuration details. SM only has a view on the edge-to-edge resources of the network omitting all details about paths, number of hops and per-hop configurations. This view is the *edge-to-edge Network Configuration* (NC), which is provided by Network Dimensioning to Service Subscription.

Edge-to-edge NC={QoS-class|ingress-egress|min-demand–sustainable throughput}

Edge-to-edge NC has a similar form to the TM. The *sustainable throughput* is calculated by the TE algorithms and is the effective (longer-term) reserved capacity between two TE cycles. The min-demand provides enough resources such that the SLS QoS requirements are met with a "very large probability" (again defined by business policies). Therefore the difference between sustainable throughput and min-demand provides a buffer of spare resources.

Network Dimensioning, an off-line component encompassing the time-dependent aspects of TE, calculates the edge-to-edge NC based on the TM and its view on internal network resources. The interworking between Service Subscription, Traffic Forecast and Network Dimensioning is called the Resource Provisioning Cycle (RPC). The RPC may be triggered periodically, e.g. every day, or on exception e.g. when Service Subscription recognises that future subscription requests may not be accommodated within the resources given in the current cycle. Between two RPCs, the Service Subscription and Invocation modules decide on the admission control of new SLSs and service invocations based on the buffer of spare resources determined by the TE system. It is important to note that new SLSs do not trigger immediate interaction between the SLS and TE systems. Of course, new subscribed SLSs are taken into account in the next RPC for calculating the (new) network configuration.

4 Voice and multimedia over IP illustrated

This section illustrates the principles outlined above for a corporate IP VPN and an NGN architecture where an IP backbone connects a number of Trunking Gateways. The focus is (again) on the SM aspects, making an abstraction of the resource-provisioning problem. It is supposed that the resource management system of the provider is capable of providing a Virtual Wire Per Domain Behaviour (VW PDB [5]) between two edge routers. A VW in this context is a Virtual Leased Line with strict edge-to-edge delay and packet loss guarantees.

4.1 QoS-capable Virtual Private Networks

Figure 6 shows two sites of an enterprise connected over a public IP network through a Virtual Leased Line. The Customer Premises Equipment (CPE), e.g. an enterprise access router, is physically connected to an IntServ/DiffServ edge router of the provider. DiffServ provides a VW PDB between the provider edge routers, yielding a VLL between the CPE routers. It is straightforward to extend the example to a multi-edge VPN, although this is not covered in this example.

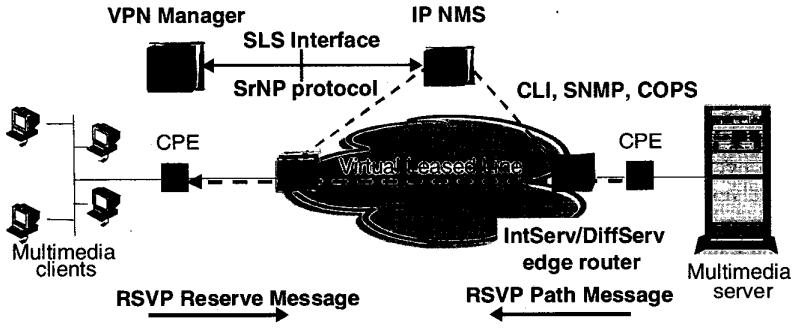

Figure 6: Multiplexing Multimedia Streams in a VLL

The IP Network Management System (NMS) handles service negotiation, subscription and (longer-term) configuration of the network through the Service and Resource Management functional entities introduced in Section 3. Individual multimedia applications are invoked through RSVP, i.e. IntServ flows are multiplexed into the VLL (and DiffServ Virtual Wire). The per-flow admission control, i.e. service invocation is situated at the provider's edge router.

The *service subscription* process is the negotiation between the customer and the provider which, if successful, will result in a SLA between the two entities. In this example the technical part of the contract - the SLS - describes the (uni-directional) VLL, offering statistical delay/loss guarantees for a well-defined (aggregate) throughput. The SLS may be conveyed in a paper contract or it may be obtained through electronic negotiation, e.g. a Web-based application as part of the SM system of the provider. We have specified a new Service Negotiation Protocol (SrNP) for this purpose (see section 5).

The result of the subscription process is a VLL for use by the company employees for video services, for example. Individual video streams within the VLL are invoked done through specific invocation requests, e.g. RSVP. This is an explicit, in-band service invocation process as discussed in the previous section. The multiplexing of the multimedia streams onto the VLL is performed at the provider's edge router. The per-flow admission control at the edge router consists in checking whether the resource-request of the new video stream (signalled by the RSVP-reserve message) fits in the overall throughput guarantee of the VLL, considering the existing traffic on that VLL.

In this example, the per-flow admission control is performed by the provider's edge router, which requires the implementation of RSVP and IntServ/DiffServ interworking capabilities. This is an extra service offered by the ISP, which could also be done by the company itself at the CPE. If the latter case, the provider acts as a "pure DiffServ" operator, offering bandwidth pipes for aggregate packet streams. The provider edge routers are unaware of individual media flows and – from the provider viewpoint – the service invocation process is obviated.

4.2 Connecting Trunking Gateways

The second example deals with the transport of voice calls over an IP backbone network by interconnecting Trunking Gateways (Figure 7). IP transport providers (ISPs) sell transport connectivity services to application service providers (ASPs), e.g. voice providers. The voice provider has a restricted number of Trunking Gateways (GWs), signalling gateways and a central call server, the Media Gateway Controller. The GWs are physically connected to DiffServ edge routers, which themselves are logically interconnected by DiffServ Virtual Wires.

As in the previous example, the IP NMS of the ISP contains SM and TE functionality. However, the dynamic per call handling, traffic conditioning and service invocation logic, is now handled by the boxes of the ASP (Media Gateway Controller and Trunking Gateways).

The SLA (between ISP and ASP) outlines the number of GWs, the expected load between each pair of GWs, i.e. capacity of the trunks, and the maximum edge-to-edge delay of each trunk. The estimation of the required trunk size is the

responsibility of the voice provider and might be based upon different techniques such as the Erlang-B type of dimensioning calculus used in telephone networks. The service subscription process yields a SLA describing a multi-edge QoS-capable IP VPN that will be used for transporting voice calls. This is a logical overlay network offering multimedia services to individual users (the customers of the voice provider). The technical information of the SLA, i.e. the {SLS}-set, is stored in the SM system of the ISP *and* the Media Gateway Controller of the ASP.

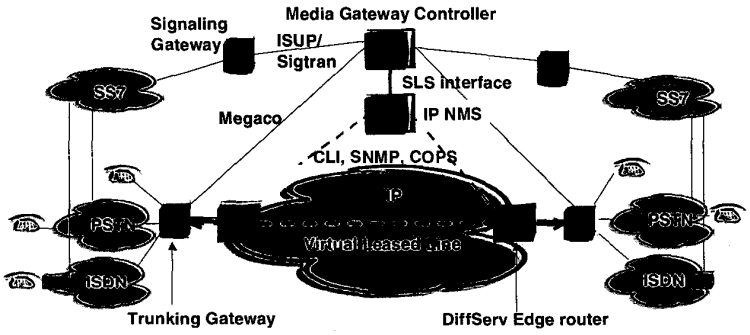

Figure 7: Connecting Trunking Gateways

The ISP's resource management system provisions the network based on all SLAs by configuring the edge and core routers under its control through TE. CLI (Command Line Interface), SNMP (Simple Network Management Protocol, [13]) or the COPS (Common Open Policy Service, [14]) protocols can be used to provide the routers with the appropriate configuration information (PHBs). This resource provisioning cycle will typically be done on a granularity of hours or more.

Admission control per multimedia flow is outsourced by the ISP to the ASP's Media Gateway Controller (MGC). The service invocation process is part of the multimedia plane functionality and is not handled by the IP transport plane. For example, the ISP's edge routers are voice-call unaware and perform the traffic conditioning on the aggregate streams as agreed upon in the SLA.

SS7 voice-call signalling is captured at the Signalling Gateway and the information is forwarded as ISUP messages over SIGTRAN (a signalling transport protocol defined in the IETF) to the media gateway controller. The latter performs per-call admission control based on its knowledge of all-ongoing calls and size of the provisioned pipes (SLA). Basically the call is admitted if the Media Gateway Controller finds out that there is still enough capacity in the VLL connecting the Gateways (*fits the 64k still in the 2 Megabit pipe*). If the call is admitted the Gateway Controller instructs the Gateways to open the gate and sends per-flow state information to the Gateways by e.g. the Megaco protocol.

5 A Proposal for a Service Negotiation Protocol: SrNP

Compared to manual service negotiation methods, through fax or email for instance, automated service negotiation offers high degrees of flexibility to the customer and provider by reducing the time to request and gain access to services. For example, electronic negotiation could enable a customer IP VPN manager module to "update" the IP VPN/VLL characteristics within certain limits previously agreed in the SLA. For example, the SLA could state that the VLL capacity is allowed to (semi) dynamically vary with 10% around a fixed average. Within a corporate VPN (Figure 6), the variation could be determined by e.g. the number of employees currently on-line. In the example of the voice provider (Figure 7), the VLL capacity could be increased depending on the number of calls blocked by the Gateway Controller. To enable this flexibility, we have specified a protocol for SLS negotiation, the *Service Negotiation Protocol (SrNP).*

SrNP applies at *subscription* times, for establishing, modifying and terminating service contracts. SrNP could also apply at service *invocation* times for implicit invocations, provided that the service contract allows this and that protocol implementation (see below) is compatible with the invocation means employed by the network, e.g. RSVP.

It should be noted that the SrNP is not specific to any SLS format, or to the context of a SLS. It is general enough to apply for negotiating any document, provided that it is in the form of attribute-value pairs (filled-form-like document). In this general model, the target of the negotiation process, operated by using SrNP, is to agree on the values of the attributes (information elements) included in the document under negotiation; and not on the information elements to be included in the document.

In the above context, SrNP provides for appropriate messages and procedures required for pursuing an agreement, thus offering the necessary primitives required to operate the particular negotiation logic (responsible for determining the terms and conditions for establishing an agreement).

SrNP is *session-oriented* and adopts a *client-server, dialogue-based* (half-duplex) approach. Specifically, SrNP operates as follows. The client issues *proposals* and the server responds by issuing *revisions* (indicating alternatives on client's proposal) or an *agreed proposal* (agreement on the last sent proposal by the client). The protocol concludes the negotiation process when the server responds with an *agreed proposal* and the client *accepts* it, or when either party *rejects* the other party's response. To ensure graceful termination, the protocol utilises a *response timer* for guaranteeing that a party cannot wait forever to receive a response from the other party.

SrNP also offers the features of 'take it or leave it' and 'please wait'. One party (the client or the server) may designate one of its responses as being its last word (*last proposal, last revision*), meaning that the other party must respond with a definite answer (*accept* or *reject*). The protocol allows for the server to *hold the proposal* i.e. to postpone its response to the client's *proposal* (e.g. should the server negotiation logic sees that an agreement is likely to be reached in the near future). In this case an explicit confirmation by the client is required (*accept to*

hold, specifying also the details of the contact point to resume the negotiation process).

A number of alternative protocol stacks can be used to realise SrNP. SrNP messages could be encoded in ASCII, BER/TLVs or XML as convenient for the stack used. Note also that it could be possible to encapsulate SrNP messages in widely deployed protocols such as RSVP (by defining new TLVs) and COPS (by specifying a new client-type). The latter is required when SrNP is to be used at invocation times.

It should be noted that the semantics and format of the document under negotiation are transparent to the protocol itself, although in this instance we assume the SLS template specified in [12].

Currently there are two implementations of SrNP; one based directly on TCP/IP and the other on HTTP. In both implementations, the SrNP messages as well as the SLA and the revised alternatives are encoded in XML.

6 Conclusions

In this paper we first proposed a layered service model for QoS-based IP networks. The model allows for a clear understanding and "mapping" of high-level IP connectivity services to low-level DiffServ PHBs through well-defined Service Level Specifications and QoS classes. Next we proposed an architectural model for supporting QoS in DiffServ networks. The model emphasises the importance of the Management plane in providing QoS and gives a functional decomposition of the main service and resource management aspects. The key concepts are the following:

The architecture introduces a two-level approach for (operational) service management and negotiation, namely *service subscription* and *service invocation*. These processes occur at different time scales. Subscription handles the longer term-based service requests such as IP VPNs, while service invocation acts on a per-call basis. This two-level approach for admission control contributes to increasing the scalability of IP backbones by enabling higher degrees of aggregation.

The architecture makes a clear distinction between the customer (SLS) aware components and the resource (QoS class) aware components. The SM system has knowledge about customers and edge-to-edge capacities and capabilities but is unaware of the internal network details. The Resource Management system knows about all network resources but only acts on (aggregate) QoS classes. The interworking is defined through the *resource provisioning cycle*.

The focus of the paper has been on the framework for the deployment of service and resource management algorithms in an overall architecture, which meets the demands of future QoS-based IP networks rather than on a detailed treatment of the specific algorithms and mechanisms to be deployed. The decomposition of the overall problem into specific functional blocks, the separation of service-related and resource-related sub-systems, and the two level approach to service negotiation and invocation are the main results.

The overall architecture enables the provisioning of QoS guarantees to individual as well as aggregate flows while maintaining a scalable solution. Ongoing development effort within the TEQUILA project aims to further validate the proposed layered approach, with final results expected by mid-2002.

Acknowledgements

This work was undertaken in the context of the Information Society Technologies (IST) TEQUILA project, which is partially funded by the Commission of the European Union. The authors would like to acknowledge the implicit contributions of their colleagues in the TEQUILA project to the ideas and concepts described in this paper.

References

[1] R. Braden, et al., "Integrated Services in the Internet Architecture: An Overview", RFC 1633, 1994.

[2] R. Braden et al, "Resource Reservation Protocol - Version 1 Functional Specification", RFC 2205, 1997.

[3] S. Blake, et al, "An Architecture for Differentiated Services", RFC 2475, 1998.

[4] K. Nichols, S. Blake, F. Baker, D. Black, "Definition of the Differentiated Services Field in the IPv4 and IPv6 Headers", RFC 2474, 1998.

[5] K. Nichols, B. Carpenter, "Definition of Differentiated Services Per Domain Behaviours and Rules for their Specification", RFC 3086, 2001.

[6] K. Nichols, V. Jacobson, L. Zhang, "A Two-bit Differentiated Services Architecture for the Internet", RFC 2638, 1999.

[7] The TEQUILA IST Project. Website: http://www.ist-tequila.org/

[8] P. Trimintzios et al, "A Management and Control Architecture for Providing IP Differentiated Services in MPLS-based Networks", IEEE Communications, Vol. 39, No. 5, pp. 80-88, IEEE, May.

[9] E. Rosen, A. Viswanathan, R. Callon, "Multiprotocol Label Switching Architecture", RFC-3031, 2001

[10] P. Trimintzios et al., "Engineering the Multi-Service Internet: MPLS and IP-based Techniques", Proc. of the IEEE International Conference on Telecommunications (ICT'2001), Bucharest, Romania, Vol. 3, pp. 129-134, IEEE, June 2001

[11] D. Grossman, "New Terminology for DiffServ", Internet Draft draft-ietf-diffserv-new-terms-04.txt, 2001.

[12] D. Goderis et al, "Service Level Specification Semantics and Parameters", Internet Draft draft-tequila-sls-01.txt – http://www.ist-tequila.org/, July 2001.

[13] J. Case, M. Fedor, M. Schoffstall, J. Davin, "A Simple Network Management Protocol (SNMP)", RFC 1157, 1990

[14] K. Chan et al, "COPS Usage for Policy Provisioning (COPS-PR)", RFC 3084, 2001