An Architectural Framework for Providing QoS in IP Differentiated Services Networks

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

D. Goderis, Y. T'Joens
Alcatel Corporate Research Center
Fr. Wellesplein 1, 2018
Antwerpen, Belgium
{Danny.Goderis, Yves.Tjoens}@Alcatel.be

P. Georgatsos
Algosystems S.A.
Sardeon Str. 4, 172 34
Athens, Greece
pgeorgat@algo.com.gr

L. Georgiadis
School of Electrical and Computer Eng.
Aristotel Univ. of Thessaloniki
PO Box 435, 54006, Thessaloniki, Greece
leonid@eng.auth.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

R. Egan
Thales Research
Worton Drive,
Worton Grange Industrial Estate
Reading, Berkshire RG2 OSB, U.K.
richard.egan@rrl.co.uk

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

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

Abstract

As the Internet evolves towards the global multi-service network of the future, a key consideration is support for services with guaranteed Quality of Service (QoS). The proposed Differentiated Services (DiffServ) framework, which supports aggregate traffic classes rather than individual flows, is seen as the key technology to achieve this. DiffServ currently concentrates on control/data plane mechanisms to support QoS but also recognises the need for management plane aspects through the Bandwidth Broker (BB), though the latter has not yet been fully addressed. In this paper we propose a model and architectural framework for supporting end-to-end QoS in the Internet through a combination of both management and control/data plane aspects.

Within the network we consider control mechanisms for Traffic Engineering (TE) based both on explicitly routed paths and on pure node-by-node layer 3 routing. Management aspects include customer interfacing for Service Level Specification (SLS)

0-7803-6719-7/01/$10.00 (c) 2001 IEEE

negotiation, network dimensioning, traffic forecasting and dynamic resource and routing management. All these are policy-driven in order to allow for the specification of high-level management directives. Many of the functional blocks of our architectural model are also features of BBs, the main difference being that a BB is seen as driven purely by customer requests whereas, in our approach, TE functions are continually aiming at optimising the network configuration and its performance. As such, we substantiate the notion of the BB and propose an integrated management and control architecture that will allow providers to offer both qualitative and quantitative QoS-based services while optimising the use of underlying network resources.

Keywords

Traffic Engineering, Differentiated Services, SLS, end-to-end QoS, IP Management.

1 Introduction

With the prospect of becoming the ubiquitous all-service network of the future, the Internet needs to evolve to support services with guaranteed Quality of Service (QoS) characteristics. This has prompted the research community to devise a number of approaches for providing QoS to Internet applications. In recent years the Internet Engineering Task Force (IETF) has proposed the Differentiated Services (DiffServ) [1] model, which has been conceived to provide QoS in a scalable fashion. Instead of maintaining per-flow soft state at each router, packets are classified, marked and policed at the edge of a DiffServ domain. A limited set of Per Hop Behaviours (PHBs) differentiate the treatment of aggregate flows in the core of the network, in terms of scheduling priority, forwarding capacity and buffering. Service Level Specifications (SLSs) are used to describe the appropriate QoS parameters the DiffServ-aware routers will have to take into account, when enforcing a given PHB. Thus micro-flow-based treatment is restricted at the DiffServ domain border while the transit routers deal only with aggregate flows, according to the DiffServ Code-Point (DSCP) field of the IP header. This procedure leads to the provision of coarse-grained QoS to applications, in a qualitative instead of a quantitative fashion, although quantitative QoS guarantees could also be provided, using for example the Expedited Forwarding (EF) PHB.

In emerging multi-service telecommunication networks (e.g. based on ATM technology) there is a hard sense of QoS. This is defined in terms of delay, jitter, throughput, service availability, or any other service specific metric that is applicable. To achieve such high QoS guarantees, control plane mechanisms are used to reserve resources on demand but management plane mechanisms are also used to plan and provision the network and to manage requirements for service subscription according to available resources [2]. DiffServ has so far concentrated in control plane mechanisms for providing QoS. However, it would not seem possible to provide QoS without the network and service management support, which is an integral part of QoS-based telecommunication networks.

Considering in particular the DiffServ architecture (see Figure 1), one of the most challenging issues is end-to-end QoS delivery. The DiffServ architecture suggests only ingredients/mechanisms for QoS delivery i.e. mechanisms for relative packet forwarding treatment to aggregate flows and mechanisms for traffic management and conditioning;

by no means does it suggest an architecture for end-to-end QoS delivery. In order to provide end-to-end quantitative service guarantees, DiffServ mechanisms should be augmented with intelligent Traffic Engineering (TE) functions.

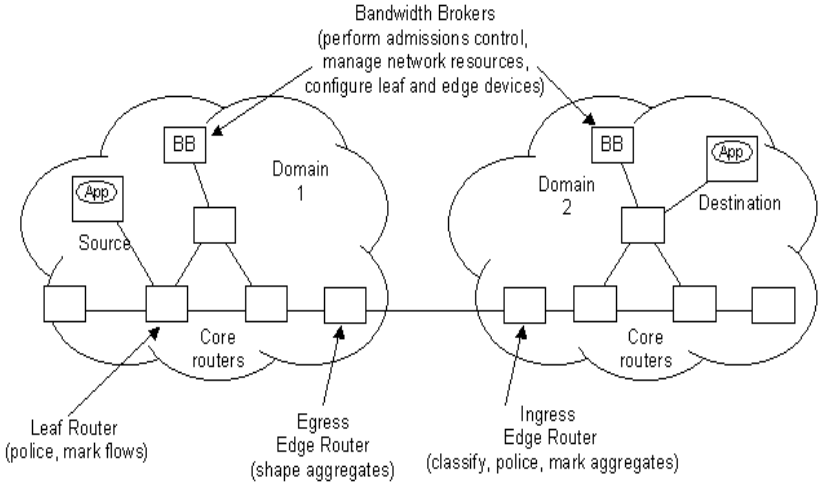

Figure 1: The DiffServ Architecture.

TE is in general the process of specifying the manner in which traffic is treated within a given network. TE has both user and system-oriented objectives. The users expect certain performance from the network, which in turn should attempt to satisfy these expectations. The expected performance depends on the type of traffic that the network carries, and is specified in the SLS contract between customer and Internet Service Provider (ISP). The network operator on the other hand should attempt to satisfy the user traffic requirements in a cost-effective manner. Hence, the target is to accommodate as many as possible of the traffic requests by using optimally the available network resources.

Multi-Protocol Label Switching (MPLS) [3], is an important emerging technology for enhancing IP in both features and services. Although, the concept of TE does not depend on a specific layer 2 technology, it is argued that MPLS [4] is the most suitable tool to provide TE by setting up of explicit routes.

With the advent of IP DiffServ and MPLS, the area of IP TE has attracted a lot of attention in the last couple of years. [5], [6], and [7] are a few of the most recent projects in this area. The TEQUILA project (Traffic Engineering for Quality of Service in the Internet, at Large Scale]) is one of them. The objective of TEQUILA is to study, specify, implement and validate a set of service definition and TE tools in order to obtain quantitative end-to-end QoS guarantees through careful dimensioning, admission control and dynamic resource management of DiffServ networks.

In this context, the paper discusses the issues in this area, considering both intra and inter-domain operations, and proposes a functional framework towards a complete solution for end-to-end QoS in the Internet. We take the position that the future Internet should offer a *variety* of service quality levels ranging from those with explicit, hard

performance guarantees for bandwidth, loss and delay characteristics down to low-cost services based on best-effort traffic, with a range of services receiving qualitative traffic assurances occupying the middle ground. Assuming a DiffServ and/or MPLS IP-based network infrastructure, we propose a functional architecture for TE specifying the required components and their interactions for end-to-end QoS delivery. The starting point is the specification of SLSs agreed between ISPs and their customers, and their peers, with confidence that these agreements can be met. The SLSs reflect the elemental QoS-based services that can be offered and supported by the ISP. The specified SLSs set the objectives of the TE functions, these being fulfilment and assurance of the SLSs. To this end, the proposed framework ensures that agreed SLSs are adequately provisioned and that future SLSs may be negotiated and delivered through a combination of static, quasi-static and dynamic TE techniques *within* domains and on an *inter-domain* basis. It proposes solutions for operating networks in an optimal fashion through dimensioning and subsequently through dynamic management functions ("*first plan, then take care*").
The rest of the paper is organised as follows: Section 2 proposes a SLS template and describes its contents and semantics. Some example instantiations of SLSs are also presented to illustrate the potential use of this template. In Section 3, we present a functional architecture as developed in the TEQUILA project, aiming at providing end-to-end QoS in the Internet. Each of the main functional components is examined in detail. Examples on TE are used in Section 4 as a first means to validate this architecture. Section 5 discusses realisation issues pertaining to the proposed architecture. Finally in Section 6 we present a summary and point to our future research work in this area.

2 Service Level Specifications

2.1 Motivation for a standardised set of SLS parameters

In this section we aim to substantiate the notion of SLS [1] by defining the associated necessary information. Today, QoS-based services are offered in terms of contract agreements between an ISP and its customers. Such agreements, and especially the negotiations preceding them, will be greatly simplified through a standardised set of SLS parameters. An SLS standard is also necessary in order to allow for a highly developed level of automation and dynamic negotiation of SLSs between customers and providers. Such automation will be helpful in providing customers (and providers) the technical means for dynamic QoS provisioning. Moreover, the design and the deployment of BB capabilities [8] require a standardised set of semantics for SLSs being negotiated between both the customer and ISP and among ISPs.

Note that although we allow for a number of performance and reliability parameters to be specified, in practice a provider would only offer a finite number of services, even for those with quantitative QoS guarantees. Therefore, parameters such as delay, mean-down-time, etc. could only take discrete values from the set offered by a particular provider. While offering customers a well-defined set of service offerings, this approach simplifies the TE problem from the providers' perspective. The next section summarises the SLS contents as described in our proposed Internet Draft [9], which aims at listing and promoting a standard formalism for a set of basic SLS parameters.

2.2 Contents and Semantics

The contents of an SLS [9] include the essential QoS-related parameters, including Scope and Flow identification, Traffic Conformance Parameters and Service Guarantees. More specifically a SLS has the following fields: Scope, Flow Identification, Traffic Conformance Testing, Excess Treatment, Performance Parameters, Service Schedule and Reliability.

The *Scope* of a SLS, associated to a given service offering, uniquely identifies the geographical/topological region over which the QoS of the IP service is to be enforced. An ingress (or egress) interface identifier should uniquely determine the boundary link or links as defined in [1] on which packets arrive/depart at the border of a DS domain. This identifier may be an IP address, but it may also be determined by a layer-two identifier in case of e.g. Ethernet, or for unnumbered links like in e.g., PPP-access configurations. The semantics allow for the description of one-to-one (pipe), one-to-many (hose) and many-to-one (funnel) communication SLS-models, denoted respectively by (1|1), (1|N) and (N|1).

The *Flow Identification* of an SLS indicates for which IP packets the QoS policy for that specific service offering is to be enforced. A Flow ID identifies a stream of IP datagrams sharing at least one common characteristic: DSCP, source/destination information, application information. Setting one or more of the above attributes formally specifies an SLS Flow ID.

Traffic Conformance (TC) Testing is the set of actions which uniquely identifies the "in-profile" and "out-of profile" (or excess) packets of an IP stream identified by the Flow-ID. It is the combination of the TC Parameters and the TC Algorithm. The TC Parameters describe the reference values the traffic identified by the Flow ID will have to comply with. The TC Algorithm is the mechanism enabling to unambiguously identify all "in" or "out" of profile packets based on these conformance parameters. The following is a non-exhaustive list of potential conformance parameters: *peak rate p* in bits per sec (bps), *token bucket rate r* (bps), *bucket depth b* (bytes), *minimum MTU* - Maximum Transfer Unit - *m* (bytes) and *maximum MTU M* (bytes). An example is the token bucket algorithm based on the token bucket parameters (b, r).

An *Excess Treatment* parameter describes how the service provider will process excess traffic, i.e. out-of-profile traffic. The process takes place after Traffic Conformance Testing. Excess traffic may be dropped, shaped and/or remarked. Depending on the particular treatment, more parameters may be required, e.g. the DSCP value in case of re-marking or the shapers buffer size for shaping.

The *Performance Parameters* describe the service guarantees the network offers to the customer for the packet stream described by the Flow ID and over the geographical or topological extent given by the scope. There are four performance parameters: *delay*, *jitter*, *packet loss, and throughput. Delay* and *jitter* indicate respectively the maximum packet transfer delay and packet transfer delay variation from ingress to egress. Delay and jitter may either be specified as worst-case (deterministic) bounds or as quantiles. The *packet loss* indicates the loss probability for in-profile packets from ingress to egress. Delay, jitter and packet loss apply only to in-profile traffic. *Throughput* is the rate measured at the egress. Performance guarantees might be quantitative or qualitative. A performance parameter is quantifiably guaranteed if an upper bound is specified. The

service guarantee offered by the SLS is quantitative if at least one of the four performance parameters is quantified. If none of the SLS performance parameters is quantified, then the performance parameters "delay" and "packet loss" may be "qualified". Possible qualitative values are: high, medium, low.

The *Service Schedule* indicates the start time and end time of the service, i.e. when is the service available. This might be expressed as a collection of the following parameters: time of the day range, day of the week range, and month of the year range. *Reliability* indicates the maximum allowed mean downtime per year (MDT) and the maximum allowed time to repair (TTR) in case of service breakdown.

2.3 Service Level Specification Examples

The proposed SLS format [9] covers the Premium Service Class as defined in the Internet2-project [7]. Examples covering the quantitative Virtual Leased Line (VLL) service for real-time applications and qualitative "Olympic" services are described below.

VLL service for real-time applications

- Traffic Conformance Testing: token bucket algorithm and parameters (b, r)
- Treatment of excess traffic: dropping.
- Performance Parameters: throughput R = r, delay, loss, jitter (not compulsory).

Costumers can use this SLS for multiplexing VoIP or Videoconference micro-flows into the VLL. The admission control for the micro-flows *inside* the VLL can either be the responsibility of the customer (at the access router) or the operator (at the ingress router).

Qualitative "Olympic" services

In these services the delay or loss performance parameters are set to a *qualitative* value, i.e. "low", "medium" or "high". Neither jitter nor throughput is specified. This yields the following combinations of relative "Olympic" service classes.

Delay / Loss	Low	Medium	High
low	gold green	silver green	bronze green
medium	gold yellow	silver yellow	bronze yellow
high	gold red	silver red	bronze red

Relative guarantees can be used for differentiating traffic within a pipe (1|1) or even a hose (1|N) Virtual Private Network (VPN). For example, the gold/green class could be used for giving priority to sales people over research engineers. It may also be used for differentiating types of applications, e.g. real-time web browsing over non real-time e-mail. An SLS example is provided below:

- Scope = (1|N), i.e. a hose model.
- Flow Id = DSCP-value indicating e.g. the gold/green class.
- Traffic Conformance Testing: token bucket algorithm and parameters (b, r).
- Treatment of excess traffic: (re-) marking with e.g. the gold/red class.
- Performance Parameter: gold/green class.

3 A Functional Architecture for Supporting QoS

In order to support end-to-end QoS based on the SLSs described above, we propose the functional architecture shown in Figure 2 [10]. There are three main parts in this architecture: SLS management, Traffic Engineering and Policy Management, in addition to Monitoring and Data Plane functions. The SLS part is responsible for subscribing and negotiating SLSs with users or other peer Autonomous Systems (ASs) and it performs admission control for dynamic SLSs. The TE part is responsible for:

- Obtaining the information needed to compute QoS configuration. Such information is conveyed between the customer and the service provider during SLS negotiation, and then the Traffic Forecast (TF) and Network Dimensioning (ND) blocks process it, so that it can be used by both the Dynamic Route Management (DRtM) and Dynamic Resource Management (DRsM) blocks for the QoS path calculation.
- Establishing the QoS path that has been selected to process a request, according to the QoS information provided by the SLS.
- Maintaining the QoS paths that have been assigned for use by a given request.

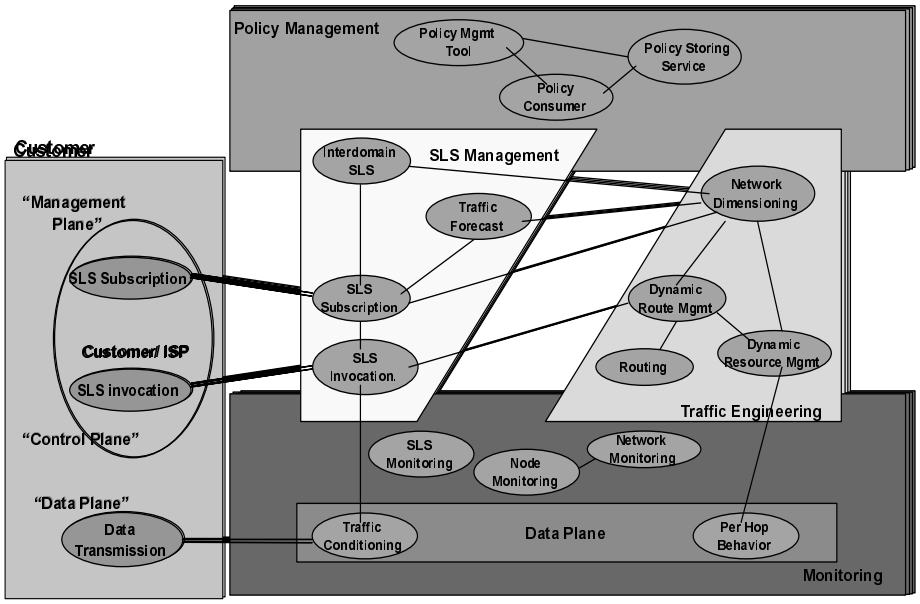

Figure 2: The TEQUILA Functional Architecture.

3.1 SLS Management

SLS Management is responsible for all SLS-related activities and is further decomposed into four sub-blocks: SLS Subscription, SLS Invocation, Traffic Forecast and Inter-Domain SLS Requester. Figure 2 shows the Interaction of the SLS management component with external customers or ISPs.

SLS Subscription is the process of customer registration and long-term policy-based admission. The customer might either be a peer Autonomous System (AS) or a business

or residential user. The subscription (or registration) concerns the Service Level Agreement (SLA), containing amongst other prices, terms and conditions and the technical parameters of the SLS. The subscription should provide the required *Authentication information* for Authentication, Authorisation and Accounting purposes when a SLS will be eventually invoked. SLS subscription contains an SLS repository with the current (long-term) subscriptions and a SLS history repository. This information serves as basic input for the *Traffic Forecast Module.* SLS Subscription performs static "admission control" in the sense that it knows whether a requested long-term SLS can be supported or not in the network given the current network configuration, this is not an instantaneous snapshot of load/spare capacity, but the longer-term configuration provided by ND. It provides a view of the current available resources to the SLS Invocation functional block.

SLS Invocation is the process of dynamically dealing with a flow and it is part of *Control plane* functionality. It performs dynamical admission control as requested by the user and this process can be flow-based. SLS invocation receives input from the SLS subscription, e.g. for authentication purposes, and has a view on the current spare resources. Admission control is mostly measurement-based and takes place at the network edges. Finally, SLS invocation delegates the necessary rules to the traffic conditioner. Both the SLS Subscription and SLS Invocation blocks are interacting with the *Inter-domain SLS Requester*, which deals with all inter-domain SLS *negotiations*, *subscriptions* and *invocations*.

The main function of *Traffic Forecast* is to generate a traffic estimation matrix to be used by the ND functional block. Traffic Forecast is the "glue" between the SLS-Customer oriented Framework and the TE Resource oriented Framework. The *input* of the Forecast module is *SLS (customer) aware* while the *output* is only *Class of Service (CoS) aware*. The *Traffic Estimation matrix* contains *per Class of Service type* (CoS), the (long-term) estimated traffic that flows between each ingress/egress pair. Its calculation is based on the SLS subscription repository, network physical topology, the physical nature and capacities of the access links, business policies, etc. A *first level of aggregation* is to bundle all the traffic that is intra-domain solely and as such a source-destination matrix will be made for each CoS type. A *second level of aggregation* takes also into account inter-domain, transient traffic and it combines all traffic that needs to end at a certain destination AS.

3.2 Traffic Engineering

This section describes in more detail the main functional blocks responsible for TE, i.e. ND, DRsM and DRtM. We explore two different TE approaches:

- **MPLS-based TE:** This approach relies on an explicitly routed paradigm, whereby a set of routes (paths) is computed off-line for specific types of traffic. In addition, appropriate network resources (e.g. bandwidth) may be provisioned along the routes according to predicted traffic requirements. Traffic is dynamically routed within the established sets of routes according to network state.
- **IP-based TE:** This approach relies on a 'liberal' routing strategy, whereby routes are computed in a distributed manner, as discovered by the routers themselves. Although route selection is performed in a distributed fashion, the QoS-based

routing decisions are constrained according to network-wide TE considerations made by the ND and DRtM algorithms. DRtM dynamically assigns cost metrics to each network interface. Route computation is usually based on shortest or widest path algorithms with respect to the assigned link costs. In order to allow for routes to be computed per traffic type or class, a link may be allocated multiple costs, one per DSCP.

3.2.1 MPLS-based TE

MPLS TE is exercised at two time scales, long-term and short-term.

- *Long-term MPLS TE* (days - weeks) selects the traffic that will be routed by MPLS based on predicted traffic loads and existing long-term SLS contracts. The routes (path, bandwidth) as well as associated router scheduling and buffer mechanisms are defined. This process is done off-line taking into account global network conditions and traffic load. It involves the global trade-offs of user and system oriented objectives and is part of the ND component.
- *Short-term MPLS TE* (minutes - hours) is based on the observed state of the operational network. Dynamic resource and route management procedures are employed in order to ensure high resource utilisation and to balance the network traffic across the MPLS Label Switched Paths (LSPs) specified by long term TE. These dynamic management procedures perform adaptation to current network state within the bounds determined by long-term TE. Triggered by inability to adapt appropriately, by significant changes in expected traffic load, or by local changes in network topology, LSPs may be created or torn down by long-term TE functions.

The long-term TE corresponds to the *time-based* capacity management functions of TE [4], whereas short-term TE corresponds to *state-dependent* capacity management functions of TE. By virtue of our model, these functions inter-operate towards a complete TE solution.

3.2.2 IP-based TE

DRtM aims at storing and maintaining the appropriate information that will be used to influence the routers' decisions. This is to enable the routers to select an appropriate route to the required destination according to the QoS parameters negotiated between the service provider and the customer.
DRtM maintains a route database for consistency purposes, and the generic architecture the IP-based TE relies upon is depicted by Figure 3, where:

1. The selection of a route is based upon the knowledge of QoS-related information, made of DSCP-related information as well as resource-specific information (expressed in terms of bandwidth, delay, jitter according to the customer's requirements),
2. DRtM embeds a PDP (Policy Decision Point, [11]), one PDP per AS typically) capability that will aim at providing the above-mentioned QoS-related information to the PEP (Policy Enforcement Point, [11]) capability embedded in the routers. Upon receipt of this information, the routing processes will have the ability to use it for selecting the best paths accordingly.

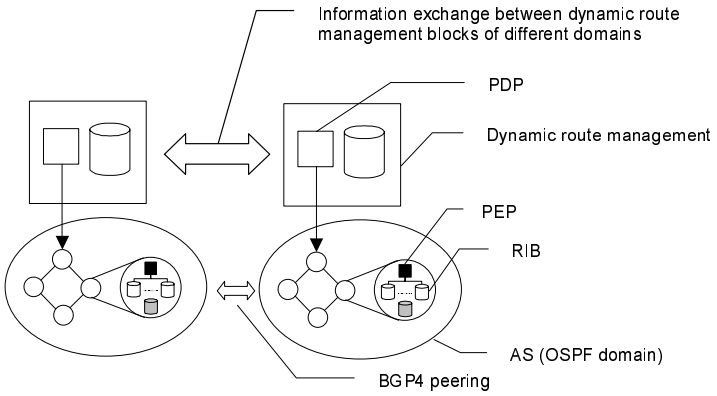

Figure 3: Interaction between the DRtM block and the IP routers.

Within a domain, OSPF routers make use of the opaque LSA (Link State Advertisement) to convey the TE-related information. It is expected that route selection at a particular router is done according to the DSCP as well as to the destination address. To achieve this, the router will maintain multiple Routing Information Base entries, one per DSCP. The OSPF algorithms embedded in the routers will be invoked upon receipt of new LSA messages, which will contain new or modified link metrics for a particular DSCP. The link metrics are defined per DSCP by DRtM, which, in turn, bases its decisions on the constraints imposed by ND. DRtM will define the link costs as seen by a particular traffic type/DSCP in such a way so that the OSPF algorithms will construct paths meeting the overall QoS objectives for that type of traffic. Whenever link costs for a DSCP need to be created, e.g. when new SLSs have been agreed with customers with QoS demands which are not currently catered for in the network, or modified, e.g. traffic conditions are such that QoS levels can no longer be met by existing routes, the PDP functionality in DRtM will notify the decision to the PEP in the routers so that new LSA messages may be generated, resulting eventually in route recalculation.

Regarding inter-domain TE, the architecture depicted in Figure 3 can be extended for conveying QoS-related information to BGP (Border Gateway Protocol) peers, thus allowing a potential enhancement of the BGP selection process. Conveying TE-related information between domains can be performed thanks to the use of an additional BGP attribute [12], which aims at reflecting the negotiated QOS requests expressed by the customer, whoever this customer may be (either an enterprise or a service provider).

3.2.3 Network Dimensioning

ND is responsible for mapping the traffic onto the physical network resources and configures the network in order to accommodate the forecasted traffic demands. ND defines MPLS Label Switched Paths (LSPs) or uses pure IP capabilities in order to accommodate the expected traffic. Traffic Forecast gets information from the current SLS subscriptions, traffic projections and historical data provided by Monitoring, and uses traffic and economic models in order to provide the appropriate forecasted traffic matrices to ND. The latter is responsible for determining cost-effective allocation of

physical network resources subject to resource restrictions, load trends, requirements of QoS and policy directives and constraints. The resources that need to be allocated are mainly QoS routing constraints, like link capacities and router buffer space, while the means for allocating these resources are capacity allocation, routing mechanisms, scheduling, and buffer management schemes. The ND component is centralised for a particular AS, although distributed implementations are also possible. In any case, it utilises network-wide information, received from the network routers and/or other functional components through polling and/or asynchronous events. This component operates in the order of days to weeks.

The main task of ND in both TE approaches considered (see sections 3.2.1 and 3.2.2) is to accept input from the Traffic Forecast Model, and depending on the particular logic of the adopted approach, to calculate and install parameters required by the elementary TE functions in the IP routers of the network. Thus, the input from Traffic Forecast Model and the output to SLS management components is assumed independent of particular logic of the TE approach employed. In the MPLS approach, ND calculates a set of routes (paths) through the network, according to the specific transfer requirements and the predicted volume of the contracted traffic (SLSs). The computed LSPs are then provided to the appropriate ingress node(s) of the network (might also be provided in the core of the network, if the MPLS capabilities of the routers allow to do so, given a label stacking capability), triggering the appropriate path establishment mechanisms.

The basic paradigm in the pure IP approach is that all TE and resource reservations should be made in a distributed fashion, thereby inherently providing high levels of scalability and fault tolerance. Hence, in this approach, ND exercises less influence on the routing decisions compared to MPLS approach, in the sense that it does not calculate explicit routes (paths) for routing specific types of traffic; this might have implications in the ability of the network to provide efficient traffic performance differentiation. In the IP-based TE approach, ND sets *administrative* parameters per link to be distributed by the IGP (Interior Gateway Protocol) employed in the network, OSPF within the context of the TEQUILA project. This enables to deterministically influence the routing decisions in the network, in the sense of indicating the cost metrics assigned per interface that will be taken into account by the route calculation process.

Two other components of the functional architecture are directly influenced by ND are the DRtM and DRsM. These to components are operating in order minutes to hours and are described in more detail in the following sections.

3.2.4 Dynamic Resource Management

DRsM has distributed functionality, with an instance attached to each router. This component aims at ensuring that link capacity is appropriately distributed between the PHBs involved in the exploitation of such resource. It does this by setting buffer and scheduling parameters according to ND directives, constraints and rules and taking into account actual experienced load as compared to required (predicted) resources. Additionally DRsM attempts to resolve any resource contention that may be experienced while enforcing different PHBs. It does this at a higher level than the scheduling algorithms located in the routers themselves. In a similar way to Dynamic Route

Management, DRsM includes a PDP capability with a corresponding PEP in the routers to actually set buffer and scheduling parameters.

DRsM is applicable to both MPLS and IP-based TE approaches. In the case of MPLS, note that LSP bandwidth is *implicitly* allocated through link scheduling parameters along the topology of the LSPs, while traffic conditioning enforced at an ingress router is used to ensure that input traffic is within its defined capacity. DRsM gets estimates of the *required* resources for each PHB from ND, and it is allowed to dynamically manage resource reservations within certain constraints, which are also defined by ND. For example, the constraints may indicate the *effective* resources required to accommodate a certain quantity of unexpected dynamic SLS invocations. Compared to ND, DRsM operates on a relatively short time-scale. DRsM manages two main resources: Link Bandwidth and Buffer Space.

Link Bandwidth: ND determines the bandwidth required on a link to meet the QoS requirements conveyed in the SLS. DRsM translates this information into scheduling parameters, which are then used to configure link schedulers in the routers. These parameters are subsequently managed dynamically, according to actual load conditions, to resolve conflicts for physical link bandwidth and avoid starving of such bandwidth for the enforcement of some PHBs.

Buffer Space: Appropriate management of the buffer space allows packet loss probabilities to be controlled. The buffers also provide a bound on the largest delay that successfully transmitted packets may experience. Buffer allocation schemes in the router dictate how buffer space is split between contending flows and when packets are dropped. According to the constraints imposed by ND for the QoS parameters associated with the traffic of a given PHB, DRsM sets the buffer space and determines the rules for packet dropping in the routers. The drop levels need to be managed as the traffic mix and volume changes. For example, altering the bandwidth allocated to a LSP may alter the bandwidth allocated for the correct enforcement of a corresponding PHB. If the loss probability for the PHB is to remain constant, then the allocated buffer space may need to change.

Through the activities of DRsM, the load-dependent metrics associated with links may change if the metrics do not reflect load directly. For example, a metric defining available free capacity in a PHB rather than used bandwidth may change when scheduling priority is increased for that PHB. For these reasons DRsM issues DRtM with appropriate updates on the state of the allocated resources for PHBs to be utilised for routing purposes. DRsM also triggers ND when network/traffic conditions are such that its algorithms are not operating effectively. For example, link partitioning is causing lower priority/best effort traffic to be throttled due to excessive high priority traffic and these conditions cannot be resolved within the constraints previously defined by ND.

3.2.5 Dynamic Route Management

DRtM is responsible for managing the routing processes in the network according to the guidelines produced by ND on routing traffic according to QoS requirements associated to such traffic (contracted SLSs).

In the MPLS approach the component is responsible mainly for managing the parameters based on which the selection of one of the established Label Switched Paths

is effected in the network, with the purpose of load balancing. It receives as input the set of explicit paths defined by ND and relies on appropriate network state updates distributed by the DRsM component. In addition, it informs the ND on over-utilisation of the defined paths so that appropriate actions are taken (e.g. creation of new paths). In this approach, the functionality of the DRtM is distributed at the network border routers/edges.

In addition to LSP selection management, in the MPLS-based approach the DRtM sets appropriate traffic meters in order to ensure the proper flow of traffic in the established LSPs, according to the defined LSP bandwidth and available capacity in the network. Note that the bandwidth assigned to each LSP is not explicitly allocated. It is implicitly allocated through link scheduling parameters through which the LSPs are defined, while the means to ensure that the input traffic to the LSPs is according to its defined capacity is enforced through traffic meters.

In the IP-based approach, Dynamic Routing Management acts as a PDP towards resolving QoS-based routing i.e. routing of traffic according to contracted SLS requirements. As SLSs are contracted, it receives the appropriate QoS-information required for routing (in the form of routing constraints) from ND and appropriately informs the routing processes in the network through the PEP capability e.g. to restrict the routing of specific traffic out of certain interfaces.

3.3 Policy Management

Policy Management includes functions such as the Policy Management Tool, the Policy Storing Service and the Policy Consumers. The Policy Consumers or Policy Decision Points (PDPs) correspond to their associated functional blocks, e.g. SLS related admission policies for the SLS Management block, dimensioning policies for the Dimensioning block, dynamic resource/route management policies for the Dynamic Resource and Route Management blocks, etc.

Although Figure 2 has shown a single Policy Consumer block for illustrative purposes, our model assumes many instances of policy consumers. In reality, the Policy Consumer is not a separate component but it is collocated with other functional blocks, e.g. SLS Subscription and Invocation, ND, DRtM and DRsM. Targets can be the managed objects of the associated functional block or of lower-level functional blocks. Policy consumers need also to have direct communication with the Monitoring functional block in order to get information about traffic-based policy-triggering events. Note that triggering events may be also other than traffic-related, in which case the specific functional block with which the Policy Consumer is associated typically generates them.

Policies are defined in the Policy Management Tool using a high-level language, and are then translated to object-oriented policy representation (information objects) and stored in the policy repository, i.e. Policy Storing Service. New policies are checked for conflicts with existing policies, although some conflicts may only be detected during execution time. After the policies are stored, activation information may be passed to the associated Policy Consumer.

Every time the operator introduces a high level policy, this should be refined into policies for each layer of the TEQUILA functional architecture forming a policy hierarchy that reflects the management hierarchy. As mentioned in the literature [13], it

is very difficult to support an automated decomposition of a policy without human intervention. The administrator should define generic classes of policies and provide some refinement logic/rules for the policy classes that will help the automated decomposition of instances of these classes into policies for each level of the hierarchical management system shown in Figure 2. These generated policies can be interpreted and enforced by the Policy Consumer associated with the responsible functional block.

4 Example Scenarios

In this section, we present example scenarios for IP-based and MPLS-based TE. In the case of MPLS-based TE, we give an example for both short-term and long-term TE.

4.1 IP-based TE

This example assumes a videoconferencing service to be deployed within an IP VPN between a specific set of sites. The corresponding QoS requirements will be conveyed in the SLS (see section 2.3 for examples). Upon receipt of a number of SLSs, the TF block sends the predicted traffic matrix to the ND block, which checks the availability of resources. Assuming the corresponding SLS subscription has been accepted, the ND block will invoke the DRtM block, which will check if there are already routes reaching the prefixes described in the Flow ID parameters.

If such routes already exist, the DRtM will have to check if they comply with the QoS requirements, in terms of available bandwidth, delay, etc. along the different paths. If yes, then no further action needs to be taken. It may be though necessary for DRsM parameters to be modified if the quantity of traffic for a particular class on the links along the path for that traffic will change significantly due to the additional traffic demands. If suitable routes do not already exist, then the PDP embedded in DRtM will send the appropriate "decision" message towards the Policy Enforcement Points (PEPs) embedded in the OSPF routers. These PEPs will then provide this TE-related information that may possibly yield a dynamic change of the cost metrics assigned to each interface involved in the SPF calculation according to the following procedure:

1. Assuming the OSPF domain consists of a single area (the backbone area), the routers of the domain will use the opaque LSA (Link State Advertisement) type 10 (area flooding scope LSA) to describe the TE topology, including bandwidth and administrative constraints. Such LSA messages will use the Link Type Length Value (TLV) to describe a single and numbered link.
2. As far as the DSCP value is concerned, it will be conveyed in sub-TLV type 9 associated to the above-mentioned TLV, while sub-TLV type 6 (maximum bandwidth) will depict value of the maximum bandwidth that can be used on the link in a given direction, according to the QoS requests related to the EF marking.
3. Once the TE routes have been computed and appropriately installed in the LSDB (Link State Data Base) of each OSPF router, each PEP is expected to report the LSDB update towards the PDP that will in turn update the route database maintained by the DRtM module, so that consistency is maintained.

In this example, the videoconferencing type of traffic towards a receiver will be EF-marked, and conveyed along a path that has been selected according to the shortest path

first algorithm that will take into account the information conveyed by the above-mentioned type 10 LSA message. Likewise, the DRsM module is expected to set up the EF PHB parameters accordingly.

4.2 MPLS-based TE

This example describes a simple scenario for long-term MPLS-based TE. The numbers and choices are only indicative and the intent is to outline the key ideas. We consider that TE is attempting to route the network traffic demands so that the load, i.e. ratio of link traffic to link capacity, on network links is kept as small as possible.

Let's assume that three customers are interfacing with the AS corresponding to a given ISP (see Figure 4). The following agreements have been negotiated beween the customers and the ISP. *Customer I:* A VLL service of 1 Mbps from point G to point B. *Customer II:* VoIP traffic with aggregate rate of 2 Mbps from point A to point B. The voice call quality should be high and each individual call should be subject to admission control. *Customer III:* VoIP traffic with aggregate of 1Mbps from Point C to any of the points B or D. The voice calls are not subject to admission control and only conformance to aggregate traffic rate is required. Voice quality is not expected to be high all the time, but it is expected to be higher than what would be obtained by assigning the traffic to best effort class.

Given these requirements, the following routes are defined. a) GAFB: *1Mbps* for customer I traffic, b) *AFB: .5Mbps* for customer II traffic, c) AEB: *1.5 Mbps,* for customer II traffic, d) *Tree, CE: 1Mbps, EB: 1Mbps, ED: 1Mbps*, for customer III traffic. The PHB treatment of the routes in all the routers is the following**:** a) Traffic on routes GAFB, AFB, AEB: EF, b) Traffic on tree CE-EB-ED: AF1

Regarding link scheduling and buffering in the routers, the following are specified. a) EF: Highest strict priority, enough buffers to effectively eliminate loss b) AF1: Second strict priority, buffers based on packet loss probability 10^{-3}.

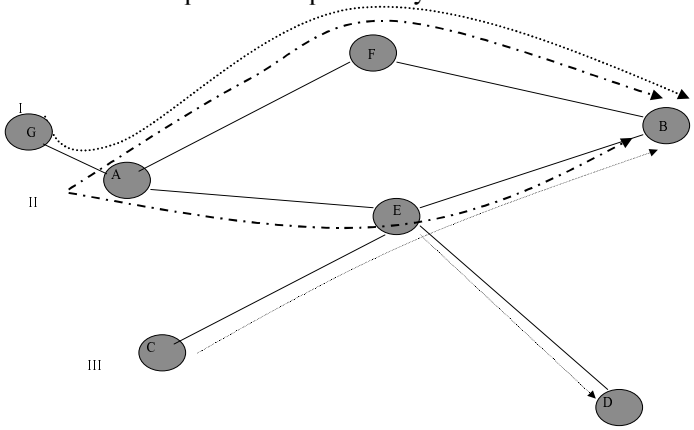

Figure 4: Example of MPLS-based Traffic Engineering.

Note that with this choice of routes the load of EF traffic on each of the links carrying this type of traffic is equalised (balanced). Note also that there are two routes for the customer II traffic. The bandwidth allocated on these routes will be used by DRtM to

perform load balancing. Finally, note that for customer III traffic, even though traffic of 1Mbps can be routed from point A to either point B or point D, this cannot occur simultaneously since the total traffic that can be injected from A cannot exceed 1Mbps. This is the reason why on link CE we have reserved bandwidth of 1Mbps (instead of 2 Mbps) and the route is effectively a tree.

We assume now that the AS system operates based on the routes and resource allocation determined by long-term MPLS TE. During the operation, however, it is found that additional EF traffic of 2 Mbps needs to be routed between nodes A and F. The short-term MPLS TE is invoked in this case in order to accommodate the increased traffic demand. We consider three possible actions that can be taken in this case.

- The new traffic can be simply routed on link AF, without taking any further actions. This results in 3.5 Mbps EF traffic on link AF, 1Mbps EF traffic on link GA, while links AE and EB have 1.5 Mbps EF traffic.
- Route the new traffic on AF, tear down route GAFB and create route GAEB for customer I traffic. This results in 2,5 Mbps of EF traffic on links AF, AE, 0.5 Mbps on link FB and 1 Mbps on link GA.
- Route the new traffic on AF, and route all customer II traffic on route AEB. This results in 3Mbps EF traffic on link AF, 2 Mbps on links AE and EB and 1 Mbps on FB and GA.

The first action is the simplest, however it results in maximum link load of 3.5 Mbps. The second results in the least maximum link load (2.5Mbps) but requires the tearing down and creation of new LSP paths. The third action results in intermediate maximal link load (3Mbps) but does not require the tearing-down and creation of new LSPs. The latter action is quicker to implement and does not cause major traffic disruption, it is therefore the appropriate action to take.

5 Realisation Issues

A system based on the model and architecture presented here addresses a spectrum of requirements: quantitative/qualitative service levels, static/dynamic SLSs, intra/inter-domain operation, MPLS and IP -based TE. It is likely that a network operator will adopt a phased approach to the deployment of a performance management solution such as the one proposed, where only a subset of the overall system is used initially.

The deployment strategy for many operators will be driven by the desire to deliver specific services for which they perceive an immediate market. Furthermore, the organisation of the Internet as a federation of autonomous systems means that intra-domain SLSs are more viable than inter-domain ones in the short/medium term at least, because inter-domain SLSs rely partly upon an agreed QoS policy among the involved service providers. A service provider with business customers might want to deploy such a system to deliver VLL and VPN services, as these could be offered on an inter-domain basis. Similarly, a service provider addressing the residential entertainment market could use TEQUILA to deliver services such as Video on Demand using only intra-domain services, provided that the content provider is directly connected to the service provider's network.

Some ISPs are already enforcing TE policies to improve the network performance, as perceived by users, and to improve the utilisation of network resources [14]. Generally, these networks support a single class of service: best effort. The current methodology for such an approach to TE involves deriving an expected traffic matrix for the network, which sometimes based on usage measurements. The traffic matrix is then passed either to an off-line Constraint-based Routing tool that calculates the QoS paths or to routers that use an online Constraint-based routing algorithm.

Deploying our proposed system in such an environment requires that services be described using the SLS template described in this paper. A new traffic matrix will be constructed containing the traffic requirements of all SLSs, both contracted and predicted, in addition to best effort traffic. The proposed network dimensioning functionality, possibly an enhanced version of the existing network design tools, will compute the required QoS paths and their bandwidth. Certain services, such as VLLs, are likely to be long-lived services and will be delivered using long term TE methods, as described above. Short-term TE methods are also supported, optimising the use of network resources and enabling the network operator to admit SLSs on a dynamic basis. If a network operator wishes to deploy this aspect of the proposed system, network monitoring functions are required as this type of SLS is only admitted if the measured traffic load on the network indicates that it is safe to do so.

6 Summary and Future Work

In this paper, we presented an architectural framework for supporting end-to-end QoS in IP DiffServ networks through a combination of management and control/data plane aspects. Within the network we assumed control mechanisms based both on MPLS explicitly routed paths and on pure node-by-node layer 3 routing. We proposed a template for Service Level Specifications (SLSs), followed by a functional architecture for supporting the QoS required by contracted SLSs, while trying to optimise use of network resources. The management plane aspects of our architecture include SLS subscription, traffic forecasting, network dimensioning and dynamic resource and route management. Most of these are policy-driven. The control plane aspects include SLS invocation and constraint-based routing while data plane aspects include traffic conditioning and PHB-based forwarding. The management plane aspects of our architecture can be thought as a detailed decomposition of the BB concept in the context of an integrated control and management architecture that aims to support both qualitative and quantitative QoS-based services.

We are currently at the stage of detailed design, in which the proposed functional blocks are specified in terms of interfaces to other blocks, behaviour and algorithms. Implementation, testing, validation and experimentation will follow. We plan to experiment with and demonstrate the system both on commercial network testbeds, based on Cisco routers, and on laboratory testbeds using Linux-based routers. We will also use a simulated testbed to validate and fine-tune the proposed algorithms and to be able to deal with large-scale networks, stress conditions, faults, etc. It should be finally stated that the proposed DiffServ-oriented management and control framework is based on similar validated work we have undertaken in the past on ATM [2] and MPLS [15].

As such, we are fairly confident that the proposed architectural framework will result in a workable solution for end-to-end QoS in a DiffServ-based Internet. We intend to report results on detailed aspects of the proposed framework in future papers.

Acknowledgements

This work was undertaken in the Information Society Technologies (IST) TEQUILA project, which is partially funded by the Commission of the European Union. We would also like to thank the rest of our TEQUILA colleagues who have implicitly contributed to the ideas presented here.

References

[1] S. Blake et al., "An Architecture for Differentiated Services", RFC 2475 (Informational), December 1998.

[2] P. Georgatsos et al., "Technology Interoperation in ATM Networks: the REFORM System", *IEEE Communications Magazine*, pp. 112-118, May 1999.

[3] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031 (Standards Track), January 2001.

[4] D. Awduche, "MPLS and Traffic Engineering in IP Networks", *IEEE Communications Magazine*, pp. 42-47, December 1999.

[5] P. Aukia et al., "RATES: A Server for MPLS Traffic Engineering", *IEEE Network Magazine*, March 2000.

[6] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, and J. Rexford. "NetScope: Traffic Engineering for IP Networks", *IEEE Network Magazine*, March 2000.

[7] B. Teitelbaum, "Qbone Architecture", 1999. www.internet2.edu/qos/wg/papers/

[8] K. Nichols, V. Jacobson, and L. Zhang, "A Two-bit Differentiated Services Architecture for the Internet", RFC 2638 (Informational), July 1999.

[9] D. Goderis et al., "Service Level Specification Semantics and Parameters", draft-tequila-sls-00.txt, November 2000.

[10] D. Goderis et al., "Functional Architecture and Top Level Design", TEQUILA Deliverable D1.1, September 2000 (For more information at: www.ist-tequila.org).

[11] D. Durham et al., "The COPS (Common Open Policy Service) Protocol", RFC 2748 (Standards Track), January 2000.

[12] C. Jacquenet, "Providing Quality of Service Indication by the BGP-4 Protocol: the QOS_NLRI Attribute", draft-jacquenet-qos-nlri-00.txt, work in progress, July 2000.

[13] J. Moffett, and M. Sloman, "Policy Hierarchies for Distributed Systems Management", *IEEE JSAC*, vol. 11, no. 9, pp. 140-141, December 1993.

[14] X. Xiao, A. Hannan, and B. Bailey, L. Ni, "Traffic Engineering with MPLS in the Internet", *IEEE Network Magazine,* March 2000.

[15] I. Andrikopoulos, et al., "Experiments and Enhancements for IP and ATM Integration: the IthACI Project", *IEEE Communications Magazine* (to appear), 2001.