Business-driven Management
of Differentiated Services

Javier Rubio-Loyola[1], Marinos Charalambides[2], Issam Aib[3], Joan Serrat[4], George Pavlou[2], and Raouf Boutaba[3]

[1]CINVESTAV Tamaulipas, [2]University College London , [3]University of Waterloo, [4]Universitat Politècnica de Catalunya

[1]jrubio@tamps.cinvestav.mx, [2]{M.Charalambides, G.Pavlou}@ee.ucl.ac.uk, [3]{iaib, rboutaba}@uwaterloo.ca, [4]serrat@tsc.upc.edu

Abstract — Several intra- and inter-domain quality of service (QoS) provisioning mechanisms for IP networks have been researched and developed using Differentiated Services (DiffServ) technology. However, the incremental efforts needed to manage DiffServ networks from a business-oriented viewpoint have received relatively little attention. This paper addresses this gap and presents a framework for achieving business-driven QoS provisioning in DiffServ over MPLS networks. We provide a rich model of SLA and business indicators as well as reasonable mapping functions that define, with key degrees of importance, the impact of business indicators over service management policies. Business and service indicators are used to control static and dynamic admission of services. The paper advances the state of the art by considering the influence of business-level objectives on the policy refinement process. We evaluate and discuss the effectiveness of our approach through a simulation environment that we developed over OPNET.

Keywords: business-driven management, DiffServ, QoS management, policy, Business Indicators

I. INTRODUCTION

Differentiated Services (DiffServ) have been proposed as a scalable approach for providing Quality of Service (QoS) in IP networks. The core philosophy is grouping traffic with similar QoS requirements into a limited number of service classes, allocating bandwidth to these classes, and differentiating their forwarding treatment throughout the network. While different QoS levels can be provided, the absence of advanced control (based on, for example, pricing, service admission, etc) can result into resource starvation and network congestion.

A Service Level Agreement (SLA) is a contract signed between two or more parties relating to a service relationship that sets a clear measurable common understanding of the role each party plays [1]. A party role represents a set of rules defining minimal service level expectations and service level obligations it has with other roles and at which constraints. Constraints normally include contract scope (temporal, geographical, etc.), the agreed upon billing policies, as well as the expected behaviour in case of abnormal service operation [1]. SLA violations, due for example to network congestion incur negative impact on the business value of a service provider and need to be minimized.

Management plane functionality [14] is needed to support end-to-end QoS based on Service Level Specifications (SLSs). Network congestion prevention and resolution, service subscription and invocation control, and dynamic traffic engineering functions have been the centre of study in intra-

domain [20] and inter-domain [12] management solutions. Although these solutions demonstrate how services can be delivered with QoS guarantees, the incremental efforts to elevate their business value have remained largely unexplored.

This paper addresses the need to bridge the gap between business value and configuration management. It provides a comprehensive review of the necessary elements to realize business and QoS management for DiffServ networks, and describes an integrated framework that eases the analysis of business-driven DiffServ management strategies. In contrast to prior work that applied business-driven techniques to drive IT and SLA management for data centres [3][5][7], we aim here to tackle the issue of QoS management in DiffServ/MPLS networks. We describe an approach to control relevant business indicators (BIs) and evaluate their relationships with service management objectives and policies. We focus on static and dynamic Admission Control (AC) of service subscriptions and extend previous work on policy refinement [18] by considering the influence of BIs when generating appropriate configuration policies. This is realized by defining a set of mapping functions that take into account the impact of BIs over service management policies. Administrator assigned weights of importance are given to each BI and are then used to derive appropriate policy parameters. We demonstrate, through simulation scenarios, the effectiveness and practicality of this approach, highlighting the issues necessary for its realization.

The paper proceeds as follows. Section II presents background information on DiffServ QoS management. Section III describes our business-driven framework for QoS management in DiffServ/MPLS networks. Section IV analyzes the business-driven policies. Section V provides experimental analysis and Section VI reviews related work. Finally, Section VII concludes with insights on future work.

II. DIFFSERV QOS MANAGEMENT

A. DiffServ QoS Management

QoS provisioning in DiffServ networks involves a range of management operations, from traffic engineering and Admission Control (AC), to dynamic management of resources. Several frameworks have been proposed for this purpose that mainly stemmed from European collaborative research projects including TEQUILA [20], MESCAL [12], and ENTHRONE [6]. All frameworks propose the use of a general model depicted in Fig. 1, where the QoS management goals are realized by three distinct functional blocks: *Service*

Management, *Traffic Engineering*, and *Policy Management*. The first is responsible for agreeing the customers' or peer domain's QoS requirements in terms of *Service Level Specifications* (SLSs). Traffic Engineering fulfills contracted SLSs by deriving network configuration. Policy Management guides the behavior of the two aforementioned blocks with a set of policies to reflect the high-level goals and objectives.
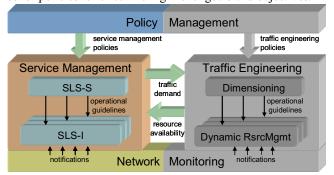

Fig. 1. Policy-driven QoS Management Framework. The business-driven approach focuses on the left part of the figure.

The business-driven approach presented in this paper focuses on the functionality provided by the Service Management block. This is realized by the *SLS-Subscription (SLS-S)* and *SLS-Invocation (SLS-I)* components. The former has a centralized off-line functionality and performs admission control on subscription requests based on resource availability provided by the Traffic Engineering system, whereas SLS-I is distributed across ingress routers and performs dynamic invocation of already subscribed SLSs based on the network runtime state, following operational guidelines provided by SLS-S. Both components assume an MPLS-enabled network.

Another issue relevant to the business-driven approach is the identification of incidents affecting the business value. Incidents like "max contracted service rate crossed" or "network link down" highlight situations that would impact the business, and for which prompt actions may be necessary.

B. Static AC for DiffServ QoS Management

The policy-based subscription logic performs static admission control of the number and type of service subscriptions in order to avoid network overloading while maximizing subscribed traffic. The policies to achieve this objective [9] are shown in Table I.

Actions P1.1 and P1.2, use the notion of service satisfaction and quality levels [13] to set the relevant parameters per *QoS Class* (QC). Satisfaction parameters define factors $\in [0,0.5]$ to derive the rates at which a service is considered *almost* (AS) or *fully* satisfied (FS). They operate on an initial average rate (SR_{AVG}) per service type that is suggested by the provider. An increasing AS factor produces lower rates ($SR_{AVG}*(1-Fctr_{AS})$). An increasing FS factor results to higher rates ($SR_{AVG}*(1+Fctr_{AS})$). The *Overall Quality Level* parameter ($OQL \in [-1,1]$) is analogous to the confidence level with which an SLS enjoys the agreed QoS. High priority QCs have an OQL close to 1.

ID	Policy action	Description
P1.1	setAlmstSatisf(QC, Fctr$_{AS}$) setFullSatisf(QC, Fctr$_{FS}$)	Sets the almost and full satisfaction factors per QC
P1.2	setQltLvl(QC, OQL)	Sets the overall quality level per QC
P1.3	setUpprLmt(TT, SU)	Sets the RAB upper limit per TT
P1.4	setACStrg(QC, TD)	Sets the AC strategy for accept/reject decision per QC

Resource availability is tracked using the *Resource Availability Buffer* (RAB), which maintains the aggregate demand of subscribed SLSs per *Traffic Trunk* (TT). As illustrated in Fig. 2, the buffer up to R^{w}_{min} can be used with high confidence even at times of congestion, whereas the area between R^{w}_{min} and R_{max} is risky because the network cannot provide guarantees [13]. Section V-A shows how the values of R^{a}_{min}, R^{w}_{min} and R_{max} are derived. Policy action P1.3 (Table I) sets the upper limit – *Subscription Upper* (SU) – as a percentage of the buffer between R^{w}_{min} and R_{max} for accepting new subscriptions and defines the level of associated risk in satisfying the QoS requirements.

Fig. 2. The resource availability buffer (RAB)

Upon a new subscription request, the overall *Traffic Demand* (TD) for a TT is updated by policy action P1.4 (Table I) taking into account the rate of the candidate service. The subscription is accepted only if $TD \leq SU$ in the RAB. Based on the satisfaction factors set by policy P1.1, TD can range from TD_{min} or TD_{max}, which are derived from AS and FS factors respectively. For the same SU, the use of TD_{min} results in more accepted subscriptions than with TD_{max}, due to the lower service rate.

C. Dynamic AC for DiffServ QoS Management

In contrast to the static nature of the subscription component, the service invocation logic is based on run-time events to regulate traffic entering the network. The policies used here are shown in Table II and are enforced on SLS-I. They perform dynamic admission control on the number of active services, as well as on the volume of admitted traffic.

Policy action P2.1 defines a threshold that signals *Target Critical Levels* (TCL) of traffic and $\in [R^{a}_{min}, R_{max}]$. The closer it is to R^{a}_{min} the earlier a notification is issued, whereas values close to R_{max} result in delayed proactive actions.

The run-time operation of SLS-I is triggered by TCL crossing alarms, which activate policy actions P2.2 and P2.3 for adjusting the *Service Rate* (SR) and the dynamic *Admission Control Threshold* (AC_{th}) of a TT. $AC_{th} \in [R^{a}_{min}, R_{max}]$ controls invocations of already subscribed services. The lower it is, the less the chances are of an incoming SLS being successfully invoked. A new service request will be accepted only if the current utilization of the relevant TT together with

the average rate of that service does not exceed AC_{th}. $SR \in$ [AS, FS] is the level of satisfaction allocated to a TT. The policy below encodes action P2.2 into policy specification following the Ponder format [10] and sets SR to the almost satisfied level upon an upward TCL threshold crossing event.

```
inst oblig /policies/sls-i/P2.2 {
      on    TCLAlarmRaised(up, TT);
      subj  s = sls-iPMA;
      targ  t = sls-i/servAdjustMO;
      do    t.setSR(tt1, sr_AS);
      when  duration(08:00-18:00);
}
```

TABLE II
SLS-I POLICY ACTIONS

ID	Policy action	Description
P2.1	setTCL(TT, TCL)	Sets the target critical level threshold wrt RAB per TT
P2.2	setSR(TT, SR)	Sets the service rate of a TT
P2.3	setACth(TT, AC)	Sets the admission control limit wrt RAB per TT

III. BUSINESS-DRIVEN DIFFSERV QOS MANAGEMENT FRAMEWORK

This section provides a business-driven QoS management framework for DiffServ/MPLS network providers. First we consider the types of SLAs to which this model is restricted.

A. SLA Template

We define an SLA S_i for customer i as a package of network services $s_i^1, \ldots, s_i^k$ offered as a bundle during a specified period of time with specific quality parameters.

S_i = {type, schedule, service package = { $s_i^1, \ldots, s_i^k$ }, availability, quality, quality degradation refund policies.}

A network service (FTP, VoIP, Web, etc.) has the form:

{service type, unitary cost, throughput [average, min, max, window], delay [average, min, max, window], ingress points, Egress points, availability, quality degradation refund policy.}

The unitary cost is the cost per month for using the service. The quality of the offered service is measured based on parameters related to throughput and delay. Service availability and average throughput over a window of time are generally the most relevant parameters. Delay is important for real-time services, such as VoIP. Other services may require more stringent parameters such as minimum delay.

A *Quality Degradation refund Policy* (QDP) can be specified at the service level and/or the SLA level. It is a rule which specifies an action to be taken by the service provider in case the offered service quality (availability, throughput, or delay) differs from the one promised.

B. Model of Service, SLA, and Business Indicators

As illustrated in Fig. 3, the overall *business utility* U_b of the set of offered services S to the network operator is modeled as a function of overall monetary profit P, services utilities U_s, and market size M_s. *Monetary profit* P is the difference of

revenue R_s from contracted services and overall operations cost C. M_s is an estimate of future market size based on repurchase intention of current customers, data gathered from competitor providers, as well as input from service usage trends. The utility U_s of an offered service S is derived from how satisfied the customers of the service were ($C_s \in [0,1]$), the profit P_s generated from that service, as well as how trendy (T_s) it is compared to others.

$$U_b = w_p P + w_{su} \sum_{s \in S} U_s \times M_s \tag{3.1}$$

$$P = \sum_{s \in S} P_s = \sum_{s \in S} R_s - C \tag{3.2}$$

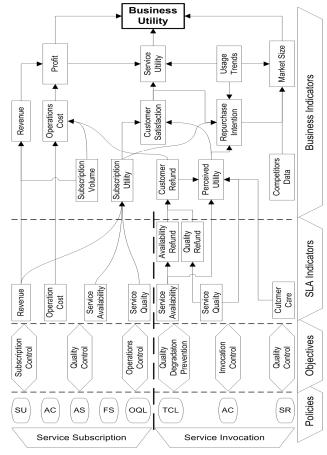$$U_s = w_{cp} \times C_s P_s + w_t \times T_s \tag{3.3}$$

Fig. 3. Model of Service, SLA, and Business Indicators.

Customer Satisfaction C_s for service s is derived from the perceived utility U_s^p of s and the expected one U_s^e. *Expected utility* U_e is function of contracted service quality, availability av_s, and cost r (revenue from the provider's perspective). *Service quality* is directly related to key QoS parameters. We restrict our evaluation of quality to service throughput th_s and delay d_s. The *Perceived Utility* U_p relies on the same parameters in addition to input from customer care C_{cr} quality and refunds of defective sessions C_{rf} if any.

C. Service Management Objectives

The *policy refinement process* relies on a set of high-level objectives specific to the service subscription and invocation management components (Fig.3). In the context of SLS-S, two objectives are considered: (a) controlling the subscription volume (*controlSubscVol*) and (b) controlling the service quality (*controlQlty*). The first relates to policies that set the upper limit in the RAB and those that set the static admission control strategy. Collectively, they regulate the amount of accepted subscription requests. The *controlQlty* objective relates to policies that set service rate factors and OQL and determine the quality of the provided service.

The objectives considered when refining SLS-I policies involve controlling (a) QoS degradation, (b) new invocations, and (c) active services. The first is achieved by policies setting the TCL parameter as this dictates how early proactive actions are taken to prevent quality degradation. Invocation admission control of already subscribed services is achieved by policies setting the AC threshold in the RAB. Finally, SR policies regulate the rates enjoyed by active services.

IV. ANALYSIS OF BUSINESS-DRIVEN POLICIES

This section describes the process by which policy parameters are derived from weighted BIs. The derivation process is based on the refinement approach presented in [18].

A. Policy parameters from BIs in the SLS-S component

Profit and *servSatisf* are the BIs influencing SLS-S. Their weights are used in linear mapping functions that derive the relevant policy parameters during the refinement process.

As shown in Fig. 4, the policy setting the SU limit partly achieves the *controlSubscVol* objective, which is influenced by both BIs. Setting the limit close to R_{max} can maximize the subscription volume. This can result in high profit but also in eventual network congestion – due to over-subscription – which can compromise service satisfaction. SU values close to R^w_{min} imply minimal risk of congestion but less profit due to lower subscription volumes. It is defined as:

$$SU = R^w_{min} + \frac{1+W_{11}-W_{21}}{2}(R_{max}-R^w_{min}) \qquad (4.1)$$

The *controlSubscVol* objective is also achieved by the policy that sets the TD parameter to be used in the static admission control decision. If TD_{max} is selected (conservative approach) profit suffers, since fewer subscriptions can be accepted, whereas higher satisfaction is achieved because there is confidence that services will be provided at their FS rates. The use of TD_{min} (less conservative approach) however, will result in higher profit, but lower satisfaction since there is confidence that services will only be provided at their AS rates. We define TD as follows:

$$TD = TD_{max}, \text{ when } W_{11} \le W_{21} \qquad (4.2a)$$

$$TD = TD_{min}, \text{ when } W_{11} > W_{21} \qquad (4.2b)$$

The two indicators also affect policies achieving the *controlQlty* objective. AS factors close to 0 and FS factors close to 0.5 imply increased service satisfaction due to the higher rates produced, but can result in lower profit as fewer service subscriptions can be accommodated. Opposite results are achieved with ASs close to 0.5 and FSs close to 0 because of lower service rates. The final value of individual factors is determined by averaging the result of applying the weights of the BIs, e.g. $Fctr_{AS}= (Fctr_{AS1}+Fctr_{AS2})/2$.

$$Fctr_{AS} = \frac{1}{4}(1+W_{12}-W_{22}) \qquad (4.3)$$

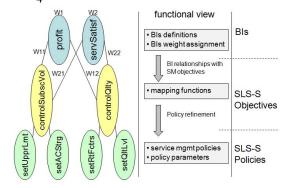$$Fctr_{FS} = \frac{1}{4}(1-W_{12}+W_{22}) \qquad (4.4)$$

Fig. 4. Relationships between SLS-S BIs, objectives, and policies

The second policy relating to the *controlQlty* objective sets OQL. Values close to 1 suggest that services are provided at their FS rates, even during congestion. This results in higher satisfaction but lower profit since a less number of services can be accommodated within a given RAB. Satisfaction is reduced with OQL values close to 0 as services are most likely to be provided at their AS rates, but an increased profit is achieved with more accepted subscriptions. OQLs close to -1 result in services with no guarantees. Satisfaction is at the lowest possible level whereas profit is at the highest.

$$OQL = (W_{22}-W_{12}) \qquad (4.5)$$

B. Policy parameters from BIs in the SLS-I component

Fig. 5 depicts the relationships of the BIs applying to dynamic service management with the associated objectives. As in the case of SLS-S, this section describes the impact of SLS-I policies on the BIs and provides the mapping functions that are used to quantify the policy parameters.

Setting TCL achieves the *control QoS degradation* objective, which is influenced by all three BIs. A TCL close to R_{max} results into delayed QoS degradation prevention actions. This can allow active services to enjoy higher than average service rates for longer and sustain a high probability of accepting new invocations. As such, service satisfaction is maximized, and loss due to invocation rejections (*invRejct*) is minimized. These conditions, however, may eventually cause network congestion and performance degradation

(*perfDegrad*) resulting into potential heavy penalties. A TCL close to R^a_{min} can have the opposite effect on satisfaction and *invRejct* loss since proactive actions are enforced too early. *perfDegrad* loss is also high in this setting because services are likely to receive less than their contracted average rates. R^w_{min} is considered the optimal TCL value for minimizing *perfDegrad* loss. This can result into average levels of service satisfaction and *invRejct* losses. Functions 4.6 to 4.11 take into account the weights of the three BIs and derive the appropriate TCL value. Two functions are provided per BI reflecting the mapping zones between R^a_{min}–R^w_{min} and R^w_{min}–R_{max}. The final TCL value is derived by using the appropriate function based on the weight, and determining the mean of the three resulting TCL instances. Fig. 6a plots the TCL mapping functions.
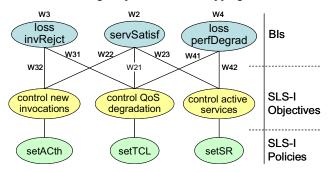

Fig. 5. Relationships between SLS-I BIs, objectives, and policies.

$$TCL_1 = R^a_{min} + 2W_{21}(R^w_{min} - R^a_{min}), \text{ when } W_{21} \leq 0.5 \qquad (4.6)$$

$$TCL_1 = (2R^w_{min} - R_{max}) + 2W_{21}(R_{max} - R^w_{min}), \text{ when } W_{21} > 0.5 \qquad (4.7)$$

$$TCL_2 = R^a_{min} + 2W_{31}(R^w_{min} - R^a_{min}), \text{ when } W_{31} \leq 0.5 \qquad (4.8)$$

$$TCL_2 = (2R^w_{min} - R_{max}) + 2W_{31}(R_{max} - R^w_{min}), W_{31} > 0.5 \quad (4.9)$$

$$TCL_3 = R^a_{min} + W_{41}(R^w_{min} - R^a_{min}), \text{ when } W_{21} > 0.5 \qquad (4.10)$$

$$TCL_3 = R_{max} - W_{41}(R_{max} - R^w_{min}), \text{ when } W_{21} \leq 0.5 \qquad (4.11)$$

The policy relating to the *control new invocations* objective sets the AC threshold, which is influenced by the *servSatisf* and *invRejct* loss indicators. Values close to R_{max} imply low probability of invocation rejections resulting to minimal losses and increased satisfaction. Low threshold values result to higher losses and less satisfaction due to the increased probability of invocation rejections – R^a_{min} represents the extreme case. Fig. 6b depicts the effect of the BIs weights on the threshold value, which can be determined by taking the mean of AC_{th1} and AC_{th2} from the functions below.

$$AC_{th1} = R^a_{min} + 2W_{22}(R^w_{min} - R^a_{min}), \text{ when } W_{22} \leq 0.5 \qquad (4.12)$$

$$AC_{th1} = (2R^w_{min} - R_{max}) + 2W_{22}(R_{max} - R^w_{min}), \text{ otherwise} \qquad (4.13)$$

$$AC_{th2} = R^a_{min} + 2W_{32}(R^w_{min} - R^a_{min}), \text{ when } W_{32} \leq 0.5 \qquad (4.14)$$

$$AC_{th2} = (2R^w_{min} - R_{max}) + 2W_{32}(R_{max} - R^w_{min}), \text{ for } W_{32} > 0.5 \qquad (4.15)$$

The last SLS-I policy involves service rate adjustments of active services, which have an impact on the user's perceived service quality and on the penalties applying as a result of performance degradation. Values close to SR_{FS} imply high levels of satisfaction and prevention of penalties. AS rates can

lead to the reverse due to unfulfilled contracted rates. The impact of the two BIs are quantified by the functions bellow:

$$SR_1 = SR_{AS} + W_{23}(SR_{FS} - SR_{AS}) \qquad (4.16)$$

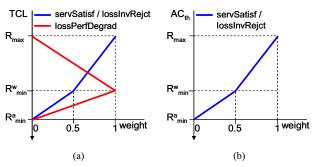$$SR_2 = SR_{AS} + W_{42}(SR_{FS} - SR_{AS}) \qquad (4.17)$$

Fig. 6. Impact of SLS-S BIs weights on (a) TCL, and (b) AC threshold.

By specifying the importance of BIs with weights and using the described mapping functions, a network can be configured according to the business objectives. An ISP may, for example, opt to minimize at the most the penalties for high revenue-generating services, or to build up its reputation through good levels of service satisfaction.

V. EXPERIMENTAL ANALYSIS

All experiments were conducted with a modified OPNET toolkit, extended to support the execution of SLS-I policies. The objective is to analyze the influence of weighted BIs on the enforcement of SLS-I policies in DiffServ/MPLS networks.

A. Services and Network Topology

We consider a network service provider that implements the business-driven DiffServ management approach described in the paper, offering SLAs with bundles of FTP, web browsing and email services. The provider classifies services into medium (medQlty) and high quality (highQlty), assigning them to AF11 and AF43 QoS classes, respectively. We consider 7 SLAs of medQlty services with average rate 3,43Mbps each (24Mbps total avg rate) and 7 SLAs of highQlty services with average rate 1,714Mbps each (12Mbps total avg rate). The network topology is shown in Fig. 7. LER1/LER4 are respectively the ingress/egress points for medium quality services and LER2/LER5 for high quality ones. Medium quality services users are located in sites 1 to 7, and sending/receiving traffic to/from site 22. High quality services users are located in sites 11 to 17, and sending/receiving traffic to/from site 23.

The above information is passed to a dimensioning process resulting to the creation of two Traffic Trunks [ingress/egress, QC, RAB_{TT}], namely TT1=[LER1/LER4, AF11, RAB_{TT1}] and TT2=[LER2/LER5, AF43, RAB_{TT2}]. Following our previous definitions, the calculation of the RAB values R^a_{min}, R^w_{min}, R_{max} considers Almost Satisfied ($Fctr_{AS}$) and Fully Satisfied ($Fctr_{FS}$)

factors and the average rates of the offered services. The provider defines $Fctr_{AS}$=0.3, and $Fctr_{FS}$=0.2 for medQlty services (served through TT1), and $Fctr_{AS}$=0.4, and $Fctr_{FS}$=0.3 for highQlty services (served through TT2). The minimum and maximum Traffic Demand is calculated with the following expressions: TD_{min}=(1-$Fctr_{AS}$)(Total avg rate) and TD_{max}=(1+$Fctr_{FS}$)(Total avg rate). This results to TD_{minTT1}=R^a_{minTT1}=16.8Mbps, TD_{minTT2}=R^a_{minTT2}=7.2Mbps, TD_{maxTT1}=28.8Mbps, and TD_{maxTT2}=15.6Mbps.
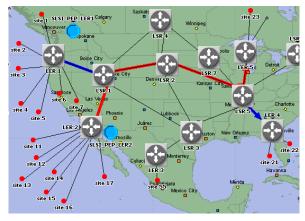

Fig. 7. Network Topology for Experimental Analysis

R^w_{min} is determined after R^a_{min} has been defined for TTs that share physical links. In this topology, TT1 and TT2 share links LSR1-LSR2 and LSR2-LSR5, which are standard E3 links with capacity 34.38 Mbps and represent the maximum resources available along the path for both TTs. To determine R^w_{min} we first determine the available spare resources after the sum of R^a_{min} of the TTs has been subtracted from the capacity of the shared link. In this case the spare resources are 10,37Mbps. These resources are split between TT1 and TT2 proportionally to their TD_{max} value, namely 6.9Mbps and 3.5 Mbps respectively. Consequently R^w_{minTT1} = 6.9Mbps + R^a_{minTT1} = 23.7Mbps. Similarly for TT2, R^w_{minTT2}= 10.6Mbps.

Finally, R_{max} in a TT is calculated by subtracting the sum of R^a_{min} for the TTs sharing resources with the TT, from the capacity of the shared link, namely R_{maxTT1} = 34.38Mbps – 7.2Mbps = 27.1Mbps. Similarly for TT2 R_{maxTT2} = 17.5Mbps.

The values of R^a_{min}, R^w_{min}, R_{max} are significantly important for the enforcement of the business-oriented SLS-I policies. Service rate and admission control adjustments are driven by these values as we show experimentally in the next sections.

B. Business-driven Reaction to Traffic Fluctuations

This section describes the enforcement of business-driven SLS-I policies due to dynamic traffic fluctuations. Consider that invocations of bundled services include the patterns of applications shown in Table III.

Consider a provider that wants to avoid losses due to invocation rejections (*lossInvRejct*) and performance degradation (*lossPerfDegrad*), for which for which he assigns

0.75 to BI weights W3 and W4 (see Fig. 5). The provider can tolerate medium levels of service satisfaction (*servSatisf*), for which a weight of 0.5 to W2 is assigned.

TABLE III
PATTERNS OF APPLICATIONS PER BUNDLED SERVICE

App	Variable	Distribution	Param.	Instances per Bundled Service
http	Page inter-arrival	Poisson	10sec	15/Medium Qty Svc 8/High Qty Svc
	Object Size	Constant	1kB	
	Image	uniform	2kB-10kB	
FTP	Inter-Request time	Poisson	15sec	15/Medium Qty Svc
	File Size	Poisson	500kB	8/High Qty Svc
email	Inter-arrival time	Poisson	15 secs	15/Medium Qty Svc
	E-mail Size	Poisson	4kB	8/High Qty Svc

Policy parameters are derived making use of the mapping functions elaborated in Section IV, and the RAB values described in Section A. For example, expressions 4.6 to 4.11 are used to derive the TCL parameter values. Provided that W2=0.5, TCL_1 is calculated with expression 4.6 (W21≤0.5). For TT1, the computation of TCL_1 results to 23,7Mbps. TCL_2 gives 25.44Mbps and TCL_3 results to 24.5Mbps with expressions 4.9 and 4.11, respectively. Taking the arithmetic mean of the three TCL values, we derive the TCL value of 24.5Mbps for TT1. The derivation of all policy parameters based on the mapping functions of Section IV.B results to the policies summarized in Table IV, which are enforced on routers LER1 and LER2 in Fig. 7.

TABLE IV
SLS-I POLICY ACTIONS

Business Indicators	Policy Actions	
W2=0,5	setTCL(TT1, 24.5Mbps)	setTCL(TT2, 12.3Mbps)
W1=0,75	setSR(TT1, 24.3Mbps)	setSR(TT2, 12.45Mbps)
W2=0,75	setACth(TT1, 24. 5Mbps)	setACth(TT2, 12.3Mbps)

Taking into account the patterns of applications per bundled services (Table III), Fig. 8 shows the throughput of TT1 and TT2, respectively, without the business-oriented policies. The Fig. 8 shows 5 minutes during which the provider serves excessive invocations. The bottom part of the figure shows the amount of invocations received and the active services during each period of the scenario execution. medQlty services are invoked every 20sec and they are active for 70sec. highQlty services are invoked every 40sec and remain active for 80 sec.

Serving excessive invocations at a time (e.g. up to 28 medQlty services, plus up to 14 highQlty services) results into network over-utilization and eventually to congestion. Without control of any kind, service satisfaction is severely reduced for both medQlty and highQlty services. In addition, potential penalties due to performance degradation should be high due to network over-utilisation and congestion during most of this scenario, severely affecting the business value of the provider.

Fig. 9 shows the behaviour of the network when driven by the business-oriented policies introduced earlier. For comparison purposes, in this execution run the patterns of the applications shown in Table III have been used, producing similar patterns of traffic as the ones in the previous experiment (Fig. 8). The number of service invocations and the period in which services are active are also the same as in the previous execution run. The actual service invocations, active services and rejected invocations during the execution of this policy-driven scenario are shown at the bottom of Fig. 9.
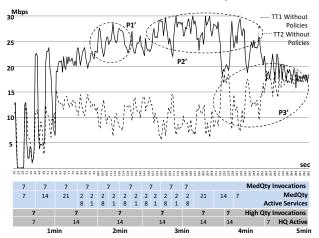

Fig. 8. Scenario execution without the business-oriented policies

The business-driven policies force the network to exploit the resources, but at the same time to protect it for eventual service degradation. For example, up to 28 medQlty services are served before sensible proactive actions are taken as a result of TCL crossings in TT1 (see P1 in Fig. 9). A total of 14 invocation rejections and service rate adjustments result to a sensible reduction of the traffic injection by users, around TCL for TT1 (see P2 in Fig. 9 compared to P2' in Fig. 8). The system has been forced to serve between 14 to 7 medQlty services, which is still highly profitable provided that the dimensioning estimations were originally set to 7 medQlty services for TT1. This way the potential losses due to invocation rejections are reduced. User satisfaction is in general acceptable as users enjoy rates close to TCL, above R^w_{minTT1} (23.7Mbps). Potential losses due to performance degradation are also low. In the execution run without policies the resources are over-exploited thus jeopardizing service satisfaction and consequently affecting the business value of the provider (see P1' and P2' in Fig. 8).

Similar results are obtained for high quality services served through TT2. An interesting situation in TT2 is presented as a result of the enforcement of TT1's policies. Due to the reduction of traffic injection in TT1, the DiffServ mechanisms allow traffic served through TT2 to be slightly increased, see for example P4 and P5 in Fig. 9. The same principle is applied in these conditions, proactive actions are taken when traffic goes beyond TCL for TT2, resulting to 14 invocation rejections of high quality services along the scenario execution and the

appropriate service rate adjustments, resulting in an eventual normalisation of traffic injection in TT2 (see P3 in Fig. 9). This way the sensible increase of traffic injection exhibited in the normal DiffServ scenario are avoided (P3' in Fig 8). The standard DiffServ mechanisms are concealed with the business oriented approach applied here to increase the business value of the network, serving between 7 to 14 high quality services and taking care of the quality of the services.
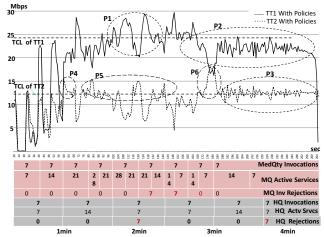

Fig. 9. Scenario execution with the business-oriented policies

In order to show the effectiveness of our approach, traffic fluctuations have been accentuated by disabling the traffic shaping mechanisms of edge routers. Also, to avoid instability due to short traffic fluctuations, we used a 5 second time window for TCL-crossings. This is exemplified in P4 in Fig. 9 where the actual policies are not triggered therein (TCL crossing<5sec).

VI. RELATED WORK

In the context of ISP upgrade and planning operations [22] investigated models linking network service performance, customer behaviour, and market dynamics to profit. Customer relations and profitability were the subject of significant research in market science and economics [16,17,21]. Work in [15] suggests strong relationships among satisfaction, perceived quality, and the disconfirmation with expected quality. [4] follows with a descriptive model relating service quality to customer repurchase intention.

SLA modelling from a policy-based enforcement perspective has been studied in [2]. In [3], a detailed business-driven SLA refinement and optimization into low-level policy configuration is presented, with a case study on incident management in utility data centres. Only raw monetary profit was used as a BI. In this paper we provided a more elaborate policy-based framework for BIs with a link to SLA and service indicators. [19] uses a policy-based dynamic mapping of application QoS parameters to DiffServ classes to provide time and location based services access control and differentiation

so as to maximize utilization and enforce differential service priorities in enterprise nomadism scenarios.

Prior work on incident management, independent of any policy-based solution, was used to assist service personnel to prioritize the processing of action-demanding quality management alerts as per provider's service level management objective [7][5]. Prioritization of service incidents based on their business impact and urgency were proposed. Our work lays down the concepts and provides proof of feasibility of business-driven management into DiffServ QoS domains, identifying the necessary elements for its realization using a rich policy-based business and SLA-driven modelling and refinement. To the best of our knowledge, no prior work addressed business-driven management in DiffServ QoS domain, from the definition of key BIs to their influence on network and service objectives, to the systematic derivation of policies and associated parameter values.

VII. CONCLUDING REMARKS AND FUTURE WORK

This paper described elements required to bridge the gap between business and configuration management in the DiffServ QoS management domain, elaborating on a bidirectional approach that eases the analysis of business-driven DiffServ management strategies.

The approach relies on business indicators that are used by ISPs to derive a network configuration that is in line with high-level business objectives. Such a holistic approach requires the consideration of both static and dynamic admission control of services. In this respect, the core contribution of the paper is the definition of mapping functions and their use by a refinement process to derive the appropriate service management policies, based on the importance of BIs and their relationship with service management objectives.

In this paper we analyzed *service satisfaction*, *profit*, and *loss* BIs. Five mappings were defined based on the influence of the first two indicators on service management objectives (SMOs). The functions control the subscription volume and the quality level of services. Similarly, twelve mapping functions have been defined based on the impact of all BIs on SMOs. They control the invocation logic, performance degradation, and the rates of currently running services. The mapping functions defined are rich enough to quantify all the necessary policy parameters of the refined policies.

The practicality and effectiveness of our approach were demonstrated with an illustrative scenario in which business-related and DiffServ networking issues were put into context.

We developed the instruments to analyse business-driven DiffServ management strategies. Future work involves the study of strategies to maximize the business value and the quality of provided services under different service usage conditions. We will also investigate the classification and validation of service subscription and invocation patterns, and their influence under different services usage patterns. Other areas that will be targeted are the study of pricing strategies and

customer frustration reactions to service degradation. These two important aspects are captured in the proposed business model but are not yet explored. We expect them to have a direct impact on service usage, service subscription and invocation for network and service providers. This may possibly require the elaboration of more complex business strategies, and possibly the involvement of realistic models of customer reactions to price and service disruptions.

REFERENCES

[1] I. Aib , N. Agoulmine, and G. Pujolle; "A Multi-Party Approach to SLA Modeling, Application to WLANs", IEEE CCNC, Jan 2005, USA.

[2] I. Aib et al.,, "Business-Aware Policy-Based Management", IEEE/IFIP BDIM'06, in conjunction with NOMS 2006

[3] I. Aib and R. Boutaba, "On Leveraging Policy-Based Management for Maximizing Business Profit", IEEE TNSM 2007

[4] E. W. Anderson and M. W. Sullivan, "The antecedents and consequences of customer satisfaction of firms," Marketing Science, 12(2), 1993.

[5] C. Bartolini, M. Salle, "Business Driven Prioritization of Service Incidents", 15th IFIP/IEEE DSOM 2004, USA

[6] E. Borcocci, et al., "Admission control algorithm for aggregated pipes service invocation in multi-domain IP environment," Int. Workshop on Next Generation Networking, Coimbra, Portugal, 2006.

[7] M. J. Buco, et al., "Utility computing SLA management based upon business objectives", IBM Systems Journal, Vol 43, No 1, 2004

[8] M. Carlson, et al., "An Architecture for Differentiated Services," RFC2475, 1998.

[9] M. Charalambides, et al., "Policy conflict analysis for DiffServ quality of service management," IEEE TNSM, 2009.

[10] N. Damianou, et al., "The Ponder policy specification language," IEEE Policy Workshop, UK, 2001.

[11] P. Flegkas et al "A Policy-based Quality of Service Management Architecture for IP DiffServ Networks," IEEE Network 16(2), 2002.

[12] M.P. Howarth, et al., "Provisioning for interdomain quality of service: the MESCAL approach" IEEE ComMag, 43(6), June 2005.

[13] E. Mykoniati, et al.,"Admission control for providing QoS in IP DiffServ networks: the TEQUILA approach" IEEE ComMag, 41(1), 2003.

[14] K. Nichols, V. Jacobson and L. Zhang, "A Two Bit Differentiated Services Architecture for the Internet," RFC2638, 1999

[15] R. L. Oliver and W. DeSarbo, "Response determinants in satisfaction judgements," J. Consumer Research, vol. 14, Mar. 1988.

[16] A. Parasuraman, et al., "A conceptual model of service quality and its implications for future research," J. Marketing, vol. 49, pp. 41–50, 1985.

[17] A. Parasuraman et al "SERVQUAL: A multiple-item scale for measuring consumer perceptions of service quality," J. Retailing, 64, 1, 1988.

[18] J. R. Loyola et al "A Methodological Approach towards the Refinement Problem in Policy-based Management Systems" IEEE ComMag, 2006.

[19] B. Tebbani, I. Aib and G. Pujolle, "SLA-based Dynamic Resource Management in Wireless Environments: An enterprise Nomadism Use Case", IJIPT journal, 3(1), pp. 13-21, 2008

[20] P. Trimintzios, et al., "Service-driven traffic engineering for intra-domain quality of service management," IEEE Network Magazine, 17(3), 2003.

[21] Y.Wang et al "An integrated framework for service quality, customer value, satisfaction: Evidence from China's telecommunication industry," Information Systems Frontiers, 6(4), 2004.

[22] Xiao J. and Boutaba R., "Assessing Network Service Profitability: Modeling From Market Science Perspective", IEEE TON, 15(6), 2007.