MESCAL

Management of End-to-end Quality of Service
Across the Internet at Large

IST-2001-37961

D1.1: Specification of Business Models and a Functional Architecture for Inter-domain QoS Delivery

Document Identifier: MESCAL/WP1/UniS/D1.1/final	
Deliverable Type: Report **Deliverable Nature:** Public	**Contractual Date:** 31 May 2003 **Actual Date:** 19 June 2003

Editor:	Paris Flegkas, UniS
Authors:	*FTR&D:* P. Morand, M. Boucadair, P. Levis, Y. Noisette, N. Cantenot *TRT:* R. Egan, H. Asgari *UCL:* D. Griffin, J. Griem *UniS:* P. Trimintzios, P. Flegkas, N. Wang, M.P. Howarth, G. Pavlou *Algo:* P. Georgatsos, T. Damilatis, G. Memenios, D. Makris
Abstract:	This document is the result of activity AC1.1 and is a key deliverable of the MESCAL project. It comprises three principal components. The first component is the business model, which defines the principal actors in QoS-based service delivery across multiple domains. The second component is the functional architecture, which identifies the key functional blocks required to support inter-domain QoS delivery. The third component is a set of three solution options that provide QoS-based services, each of which is in accordance with the functional architecture.
Keywords:	Inter-domain QoS Delivery, Business Model, Functional Architecture

France Telecom Research and Development	FTR&D	Co-ordinator	France
Thales Research and Technology	TRT	Principal Contractor	UK
University College London	UCL	Principal Contractor	UK
The University of Surrey	UniS	Principal Contractor	UK
Algonet SA	Algo	Principal Contractor	Greece

Project funded by the European Community under the
"Information Society Technology" Programme (1998-2002)

Executive Summary

This document, D1.1, is the result of activity AC1.1 and is a key deliverable of the MESCAL project. It comprises the following principal components:

- The first component is the business model, which defines the principal actors in QoS-based service delivery across multiple domains. The model is based on current business practices and draws from related models reported in the literature.
- The second component is the functional architecture, which identifies the key functional blocks required to support inter-domain QoS delivery. A functional decomposition of the architecture identifies the various functions which the network must support in each of the management, control and data planes.
- The third component is a set of three solution options that provide QoS-based services, each of which is in accordance with the MESCAL functional architecture. These solution options are a significant set of approaches for implementing inter-domain QoS, and they form the basis of the work that will be conducted in the remainder of MESCAL.

This document is structured as follows.

Section 1 is the introduction and the roadmap to the rest of the deliverable. Section 2, *The MESCAL business model*, defines the principal actors in QoS-based service delivery across multiple domains. Section 3, *Assumptions and requirements*, documents the assumptions and requirements for QoS-related services from the perspectives of both the customers and providers defined in Section 2. Section 4, *The MESCAL QoS service model*, presents the MESCAL model for Internet QoS-based services and defines the terms used in the model. Section 5, *Inter-domain QoS issues*, discusses key issues which have arisen during the development of the inter-domain QoS solution. Section 6, *The MESCAL functional architecture*, presents the functional architecture and defines the functional blocks and their interactions required to support inter-domain QoS delivery. Section 7, *Solution Space*, presents three solution options that provide QoS-based services.

The deliverable also contains Appendix A, which comprises a review of the state of the art in research, standardisation and current commercial practice in inter-domain QoS delivery.

Table of Contents

List of Figures

List of Tables

1 INTRODUCTION

1.1 The MESCAL Project

The overall objective of the MESCAL project is to propose and validate scalable, incremental solutions that enable the flexible deployment and delivery of inter-domain Quality of Service (QoS) across the Internet. MESCAL will validate its results through prototypes, and evaluate the overall performance through simulations and prototype testing. MESCAL will contribute to standardisation efforts, especially those conducted by the Internet Engineering Task Force (IETF), participate in IST clustering and actively disseminate its results.

The technical work in MESCAL is split into three work packages (WPs), following a phased approach as follows:

- WP1, *Specification of Functional Architecture, Algorithms and Protocols*, is responsible for defining business models and the generic, multi-domain, multi-service IP QoS functional architecture for inter-domain QoS delivery. Based on these models WP1 will develop algorithms and protocols for negotiation and establishment of inter-domain SLSs; and will enhance and extend inter-domain traffic engineering (TE) mechanisms and routing protocols, including the required interactions with intra-domain functionality. WP1 will also define system, subsystem and algorithm test requirements. Based on implementation experience and experimental results fed back from WP2 and WP3, later activities within WP1 will validate the initial specifications and derive enhancements as appropriate.
- WP2, *System Design and Implementation*, is responsible for undertaking basic enhancements of experimental Linux-based routers and developing simulation tools to model the general inter-domain and QoS requirements of the project. Based on the specifications from WP1, WP2 will specify the engineering approach, conduct detailed implementation design and finally implement both testbed prototypes and simulation environments.
- WP3, *Integration, Validation and Experimentation*, is responsible for configuring the required experimentation infrastructure and for conducting validation and performance evaluation activities on the prototypes and simulators developed by WP2 according to the test requirements identified by WP1. Experimentation will be executed both in the MESCAL testbed (with the support of extended development environments at other partners' premises) and using the simulators.

1.2 Role of WP1 and this Deliverable

WP1, *Specification of Functional Architecture, Algorithms and Protocols*, comprises three activities. In the first, AC1.1, *Inter-domain Business Models and System Architecture*, business models are defined and an overall functional architecture for inter-domain QoS-based services is developed, starting from the requirements, assumptions and state of the art in this area. The second activity, AC1.2, *Algorithm and Protocol Specification*, will start from the functional architecture produced in AC1.1 and will specify algorithms and protocols for: peer SLS establishment and invocation of service instances across domains; QoS enhancements to BGP; consideration of alternative, novel approaches (e.g. link state-based); integrated inter- and intra-domain SLS management and traffic engineering; multicast SLSs and traffic engineering; impact of IPv6 on traffic engineering possibilities; and information models, algorithms and protocols for an overall policy-driven system approach. The third activity, AC1.3, *Enhancements to Algorithms and Protocol Specifications*, will produce modifications and enhancements to the AC1.2 algorithms and specifications, based on feedback from simulation and implementation experience in WP2 and WP3.

This document, D1.1, is the key deliverable of activity AC1.1 of the MESCAL project. It comprises three principal components:

- A **business model**, which defines the principal actors in QoS-based service delivery across multiple domains. The model is based on current business practices and draws from similar models reported in the literature.
- The **functional architecture**, which identifies the key functional blocks required to support inter-domain QoS delivery. A functional decomposition of the architecture identifies the various functions that the network must support in each of the management, control and data planes. The top level functions constitute QoS-based planning and QoS capabilities exchange; SLS management; traffic engineering; monitoring and assurance; and traffic enforcement.
- Three **solution options** that provide QoS-based services, and each of which is in accordance with the MESCAL functional architecture. These solution options are a significant set of approaches for implementing inter-domain QoS, and they form the basis of the work that will be conducted in the remainder of MESCAL.

1.3 Structure of this Document

This document is structured as follows:

- Section 2, *The MESCAL business model*, defines the principal actors in QoS-based service delivery across multiple domains. In particular, the model defines the terms "customer" (the target recipient of QoS services) and "provider" (entities responsible for the offering and provisioning of QoS-based services).
- Section 3, *Assumptions and requirements*, documents the requirements for QoS-related services from the perspectives of both the customers and providers defined in Section 2. The requirements are drawn from current business practices and market needs as understood by the project partners. The customer requirements cover QoS characteristics, subscription, invocation, verification, and multicast requirements. The provider requirements cover QoS extension across multiple domains, efficient path discovery and negotiation, verification, scalability, resilience, incremental deployment, ease of deployment, accounting, and multicast requirements.
- Section 4, *The MESCAL QoS service model (definitions)*, presents the MESCAL model for Internet QoS-based services and defines the terms used in the model. It includes the specification of appropriate notions, entities and the relationships and associations between them, which are thought pertinent to the issue of definition and provisioning of QoS-based services in the Internet, across multiple Provider domains. The MESCAL model defined in this Section extends the model used in TEQUILA so as to cover QoS-based services that span multiple autonomous systems (ASs), rather than the domain of a single Internet Service Provider.
- Section 5, *Inter-domain QoS Issues*, identifies and discusses a number of issues related to Inter-domain QoS delivery, focusing in particular on inter-domain peering arrangements, service guarantees, traffic engineering, scalability and multicast. The MESCAL consortium partners have considered these issues during the process of developing both the MESCAL functional architecture and the solution options that implement QoS delivery in accordance with this functional architecture.
- Section 6, *The MESCAL Functional Architecture*, defines the functional architecture that will be used in MESCAL. The functional architecture consists of five top-level blocks: service planning and QoS capabilities exchange, traffic engineering, SLS management, traffic enforcement, and monitoring and assurance. These blocks are further decomposed in the Section.
- Section 7, *Solution Space*, defines three QoS-based service options, which respectively provide Loose, Statistical and Hard QoS guarantees. The Loose service option enables a provider to offer customers access to differentiated transport services. The Statistical service option provides customers access to inter-domain QoS services with firmer guarantees than

the Loose option, based primarily on qualitative guarantees. The Hard service option provides customers with inter-domain QoS services with strict performance guarantees based on quantitative levels. Section 7 then proceeds to describe three solution options that provide QoS-based services, each of these solution options corresponding to a service option, and is in accordance with the MESCAL functional architecture.

- Appendix A is a state-of-the-art review of research, standardisation and current commercial practice in inter-domain QoS delivery.

2 THE MESCAL BUSINESS MODEL

2.1 Overview

The business model assumed by MESCAL is illustrated in Figure 1. The business model depicts from the perspectives of MESCAL the stakeholders involved in the chain of QoS-based service delivery in the Internet. It is based on current business practices and draws from similar models reported in the literature e.g. by TINA-C, TMF.

It should be noted that the MESCAL business model serves only the purpose of positioning project work; it should not be taken as a complete model capturing all business roles and relationships involved in the chain of QoS service delivery in the Internet.

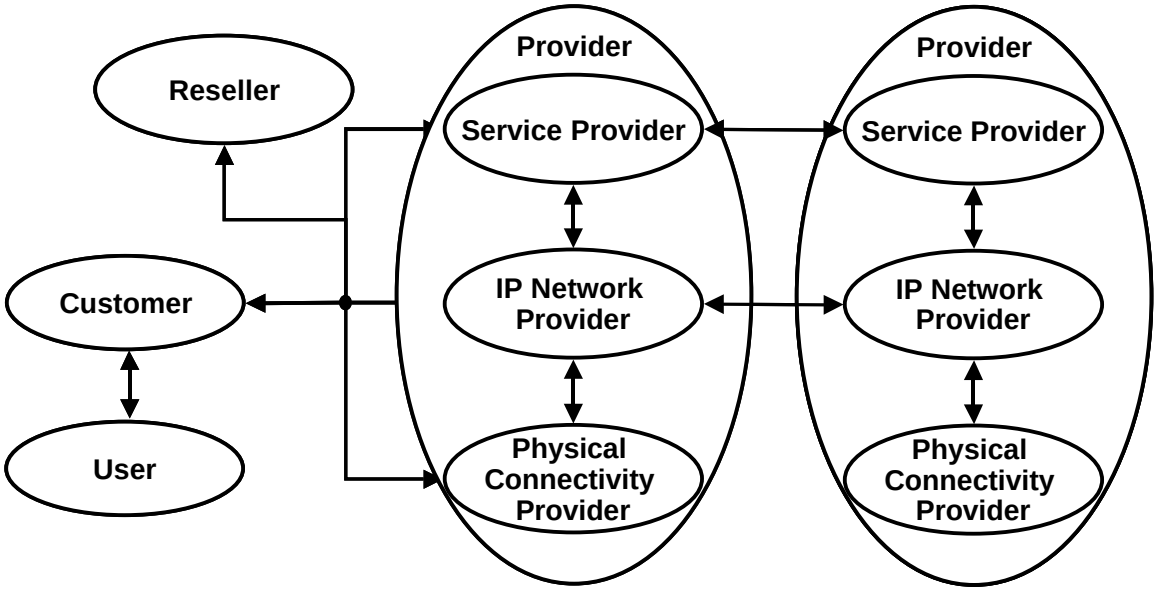


Figure 1: The MESCAL business model.

2.2 Customers and Users

A 'Customer' (subscriber) denotes an entity, which has the legal ability to subscribe to QoS-based services offered by 'Providers'. 'Customers' are the target recipients of QoS-based services. They interact with 'Providers' (or 'Resellers', see below) following a customer-provider paradigm, with the purpose to 'buy' services to meet their communication needs and requirements. Nowadays, QoS-based services are offered on the basis of respective agreements, the so-called Service Level Agreements (SLAs), setting the terms and conditions on behalf of both 'Providers' and 'Customers' for providing and requesting/accessing the services, respectively.

A 'User' is an entity (human being or a process from a general perspective), which has been named by a 'Customer' and appropriately identified by a 'Provider' for actually requesting/accessing and using the QoS-based services bought by the 'Customer'. The use of the services should be in line with the terms and conditions agreed in the SLA between the 'Customer' and the 'Provider'. In essence, 'Users' are the end-users of the services and they can only exist in association with a 'Customer'. A 'User' may be associated with more than one 'Customer' using services according to the agreed SLAs of the respective 'Customer'. For instance, an employee may be acting as a 'User' of the services that its company, 'Customer', has subscribed to, as well as a 'User' of its own subscription as a residential 'Customer'.

From the point of view of service provisioning, 'Customers' may be differentiated in terms of their size with respect to the number of geographical locations they may be present and/or the number of 'Users' they have, their type of business, the type of services they require including strictness in QoS, and the

way and the habits in requesting and using the required services. Householders, small and medium enterprises, large corporations, universities or public organisations, who are able to know to a certain or lesser degree their communication requirements, are typical examples of 'Customers'.

2.3 Providers

'Providers' are responsible for the offering and provisioning (fulfilment, assurance) of QoS-based services. Depending on the type of services offered, three types of 'Providers' are distinguished: 'Service Providers', 'IP Network Providers' and 'Physical Connectivity Providers'.

'IP Network Providers' offer QoS-based plain IP connectivity services, that is services, which provide reachability between hosts in the IP address space. Such 'Providers' must own and administer an IP network infrastructure. For connecting customers to their IP infrastructure, 'IP Network Providers' may interact with 'Access Providers' –see below. Alternatively, customers could be connected through means/facilities provided by the 'IP Network Providers' themselves. For the purpose of expanding the geographical span of the offered connectivity services, 'IP Network Providers' interact with each other, on a one-to-one peering relationship basis. This interaction spans from the physical layer to the IP network layer (thanks to the widely deployed BGP protocol) for the purpose of exchanging "Internet full routing information" (subject to relevant routing policies) and is underlined by corresponding peering agreements, based a customer-provider paradigm, with an 'IP Network Provider' acting either in a customer's or provider's role in interacting with its peers.

'IP Network Providers' may be differentiated according to the geographical span of their IP network infrastructure. As such, we may distinguish between small, medium and large 'IP Network Providers', with this distinction being relatively (to a given area size) rather absolutely defined. For example, considering a continental area, small, medium, large 'IP Network Providers' may be thought as regional (covering specific cities of a country), national (covering a specific country), continental (covering specific countries of the continent) respectively. This distinction is essential from business perspectives, as 'IP Network Providers' seek in augmenting the reachability of their services and is in line with current business practices.

Compared to the plain IP connectivity services offered by 'IP Network Providers', 'Service Providers' offer higher-level QoS-based services encompassing both connectivity and informational aspects e.g. telephony, content streaming services. As opposed to 'IP Network Providers', 'Service Providers' may not necessarily own and administer an IP network infrastructure; they need to administer the necessary infrastructure required by the provisioning of the offered services e.g. VoIP gateways, IP video-servers, content distribution servers. As such, for fulfilling the connectivity aspects of their services, 'Service Providers' may rely on the connectivity services offered by 'IP Network Providers'. In this sense, 'Service Providers' interact with 'IP Network Providers' following a customer-provider paradigm on the basis of respective agreements (SLAs). Furthermore, for expanding the geographical scope and augmenting the portfolio of the services offered, 'Service Providers' may interact with each other on a peer-to-peer or a strict customer-provider basis.

Different types of 'Service Providers' may be distinguished according to the type of the offered services e.g. VoIP, ISPs, ASPs, Content Providers. 'Service Providers' may be further distinguished according to their size in terms of customer base and/or geographical span, into small, medium and large –cf. discussion on 'IP Network Providers'.

'Physical Connectivity Providers' offer physical (up to the link layer) connectivity services between protocol-compatible equipment in determined locations. It should be noted that the connectivity services may also be offered in higher layers (layer-3 e.g. IP), however these services are mainly between specific points as apposed to the IP connectivity services offered by 'IP Network Providers' which may be between any points in the IP address space. 'Physical Connectivity Providers' are distinguished into two main categories according to their target market: 'Facilities Providers' and 'Access Providers'. These types of Providers could be seen as distinct stakeholders.

The services of 'Facilities Providers' are mainly offered to 'IP Network Providers' to provide the required link-layer connectivity in their IP network infrastructure or to interconnect with their peers as

discussed previously. As such, 'IP Network Providers' may interact with 'Facilities Providers' following a customer-provider paradigm on the basis of respective agreements (SLAs). 'Facilities Providers' may be further differentiated according to the type of technology they rely upon (e.g. optical fibre, satellite, antennas), deployment means (terrestrial, submarine, aerial) and their size in terms of geographical span and customer base.

'Access Providers' offer services for connecting 'Customer' premises equipment to the appropriate ('Service' or 'IP Network') 'Providers' equipment. They own and administer appropriate infrastructure e.g. cables, concentrators. They may be differentiated according to the type of technology they employ e.g. POTS, FR, ISDN, xDSL, WLAN, Ethernet, as well as their deployment means and their size in terms of covered geographical area and customer base. 'Access Providers' may not be present as a distinct stakeholder in the chain of QoS-service delivery. This case arises when the ('Service' or 'IP Network') 'Providers' have their own means for connecting customers to their infrastructure –either directly or through the services of other 'Providers'. In the case that 'Access Providers' appear as distinct administrative domains, they may interact at a business level, also being inter-connected, with ('Service' or 'IP Network') 'Providers' and/or 'Customers'. Interactions between 'Providers' and 'Access Providers' is mainly governed by the legislations of the established legal telecom regulation framework and may follow a customer-provider and/or a consumer-producer paradigm on the basis of respective agreements (SLAs). Traditional state-owned PNOs in Europe and CLECs in the US are typical examples of administratively distinct 'Access Providers'.

Finally, 'Resellers' are intermediaries in offering the QoS-based services of the 'Providers' to the 'Customers'. In essence, 'Resellers' offer market-penetration services (e.g. sales force, distribution/selling points) to 'Providers' for promoting and selling their QoS-based services in the market. 'Resellers' may promote the QoS-based services of the 'Providers' either 'as they are' or with 'value-added', however adhering to the SLAs of the services as required by the 'Providers'. 'Resellers' interact with 'Customers' following a customer-provider paradigm based upon respective service agreements (SLAs), and with 'Providers' based upon respective commercial agreements. Different types of 'Resellers' may be distinguished according to whether they introduce value-added or not, their market penetration means and their size in terms of points of presence and/or sales force. Dealers, electronic/computers commercial chains, service portals are typical examples of 'Resellers'.

2.4 Focus of MESCAL – Typical Business Cases

With respect to the business model presented in the previous section, Figure 2 presents the focus of MESCAL from business perspectives.

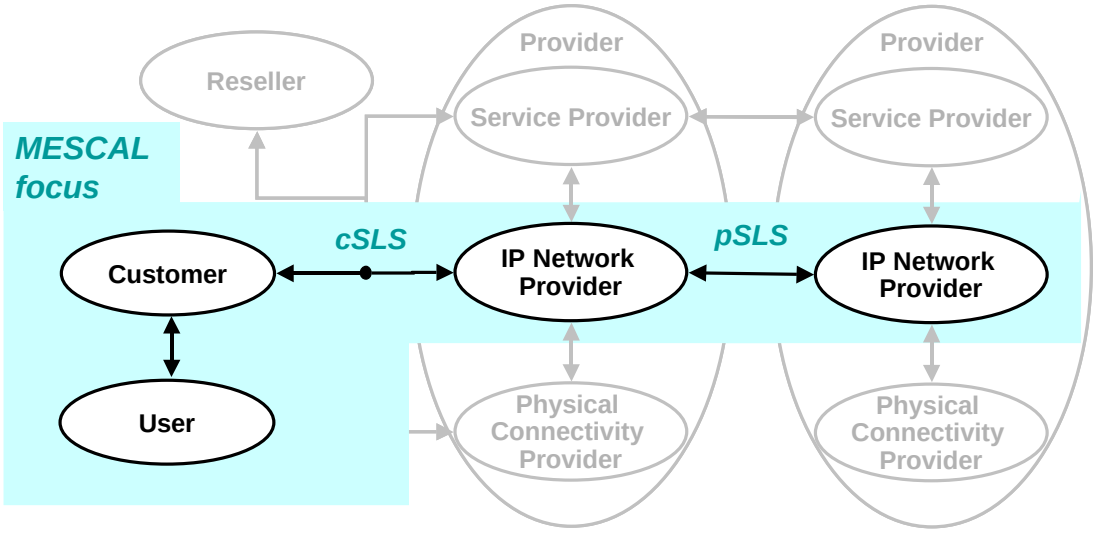


Figure 2: MESCAL focus.

MESCAL focuses on the business relationships between 'Customers' and 'IP Network Providers' and between 'IP Network Providers'. As such, MESCAL is primarily concerned with QoS-based IP connectivity services.

It is assumed that the interactions between 'IP Network Providers' and 'Access Providers' for connecting 'Customers' to the IP network infrastructure owned by the 'IP Network Providers' either do not exist (customers are connected through means/facilities provided by the 'IP Network Providers' themselves), or if they do, they are completely orthogonal to (they do not affect) the provisioning of QoS services; as such they are outside the scope of investigation.

In the above scenario, MESCAL is primarily interested in the investigation of SLAs underlying the interactions between the stakeholder domains of concern from technical perspectives (specifications, protocols etc.) and the required functionality in the 'IP Network Provider' domains for supporting these SLAs. Regarding the latter, the emphasis is on SLA fulfilment aspects through traffic engineering and service admission control functions. More specifically, MESCAL is interested in the technical aspects of SLAs, referred to as SLSs (Service Level Specifications); accounting, pricing and legal aspects are outside the scope of MESCAL investigation. The terms cSLSs and pSLSs are used to denote the SLSs between 'Customers' and 'IP Network Providers', and among peer 'IP Network Providers', respectively.

It should be noted that although the interactions with 'Service Providers' are not explicitly addressed from 'IP Network Provider' perspectives, it is believed that they are examined to a fair extent, as these interactions share some characteristics with the 'Customer'-'IP Network Provider' interactions; 'Customers' and 'Service Providers' both act in a customer role in interacting with 'IP Network Providers' –acting in a provider role.

Figure 3 depicts typical business cases exemplifying the space of applicability of MESCAL work. They present different peering cases among 'IP Network Providers' with respect to the size-based taxonomy of 'IP Network Providers' presented in the previous section -ranging from hierarchical to mesh peering cases. The validity and applicability of the MESCAL solutions will be assessed against such cases.

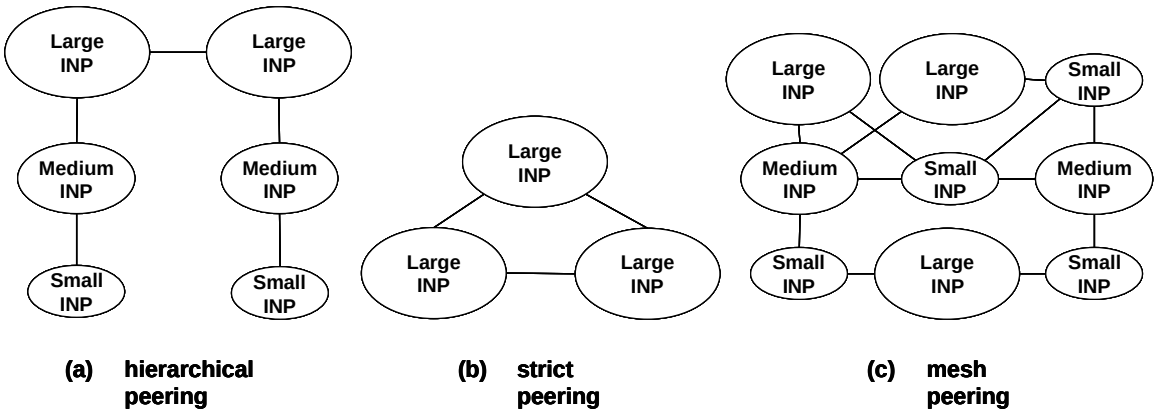


Figure 3: Typical MESCAL business cases.

The essence of the business cases depicted in the above figure is that the Internet is composed of different domains managed by INPs, which carry IP traffic of different nature and volumes, depending on the services they offer and the type of customers they serve. The purpose of MESCAL is to specify, develop and validate suitable solutions that will help the participating/peering (from an SLS standpoint) 'Providers' in provisioning a range of (value-added) IP services with guaranteed levels of quality to their end-users.

3 ASSUMPTIONS AND REQUIREMENTS

3.1 MESCAL Assumptions

"The Internet is tremendously diverse, complex, and dynamic. Nothing is ``typical''!"

Vern Paxson (IRTF chair),

The MESCAL project, which aims at deploying end-to-end quality of service at large scale, i.e. across multiple domains, relies on certain assumptions allowing a better understanding of the problem area to be addressed and thus a clear definition of the required solutions and mechanisms to be pursued.

Regarding network aspects, MESCAL assumes that providers in the Internet employ IP-based networks with DiffServ and/or MPLS capabilities for their intra-domain needs. We assume that the QoS capabilities of a given domain can be described as a limited set of well-known performance characteristics (typically one-way transit delay, inter-packet delay variation, packet loss). The QoS capabilities are tightly coupled with the constraints of the topological infrastructure. This means that the provider will engineer the network so that QoS capabilities holding from any ingress point to any egress point of its domain.

These assumptions allow the specification effort of the project to take advantage of the standardised but flexible IP DiffServ framework to build solutions for inter-domain QoS. Moreover, MESCAL considers that each provider enforces its own traffic engineering policies for its intra-domain needs.

MESCAL assumes that there is NO "Internet God". Therefore, a given provider cannot have direct service contracts with all Internet actors; Therefore, MESCAL needs to seek for solutions, which will have to rely on agreements between providers based on what is available and deployed in the different domains of the Internet.

The domain granularity assumed by MESCAL is the AS and/or set of ASs managed by the same providers. In the rest of this document when referring to interactions between ASs it may be implied to be between different providers. This is not an assumption and it does not mean that the MESCAL solution will only be applicable between ASs that belong to different administrative authorities. The MESCAL solution will be applicable even to cases where the ASs are under the same administration, i.e. provider. It is envisaged that an approach, which handles the inter-provider case, will be directly applicable to the intra-provider inter-AS (if and only if these ASs are adjacent). In the latter case some further optimisations may be applicable, and they will be studied as extensions to the general case.

MESCAL distinguishes two kinds of customers, leading to the definition of two kinds of service contracts:

- pSLS, for inter provider relationships (service-peering)
- cSLS, for end-customers (end-customer – provider relationship)

When peering, a provider wants to extend the network services it provides to its end-users within its own domain (AS scope) at a larger scale. Thus a pSLS can be viewed as a permission to send and/or receive certain quantities of traffic with contractual guarantees (destinations, throughput, QoS constraints…).

It is argued any financial settlement structure is robust only where a retail model exists that is relatively uniform in both its nature and deployment and encompasses the provision of services on an end-to-end basis [Huston99]. In MESCAL we assume the uniform financial settlement model. In this model the QoS signal initiator (i.e. cSLS) undertakes to bear the cost of the entire end-to-end traffic flow associated with the QoS signal. This is a retail model where the application initiator undertakes to fund the entire cost of data transit associated with the application.

Note that funding the entire end-to-end cost as described above does not necessarily assume a centralised model were the QoS initiator has to pay all the intermediate providers. This model is

analogous to the end-to-end retail models of the telephony, postal, and freight industries. In such a model, the participating agents are compensated for the use of their services through a financial distribution of the original end-to-end revenue, and a logical base for inter-agent financial settlements (i.e. pSLSs) is the outcome [Huston99]. Note that the service cost is out of scope of MESCAL.

As far as end- customers are concerned, a clear distinction is made between mass market and enterprise customers, as their service needs and habits differ -implying different kinds of cSLS's and, possibly, different solutions:

- On one hand, mass-market customers are known to require some quality of service “in general” (i.e. when they decide they need QoS, for any service they might want to use at the time, to/from wherever these services are offered in the Internet). Thus cSLSs in this case will rather provide loose guarantees, encompassing all possibilities that can be required by these customers.
- On the other hand, enterprise customers are known to require quality of service for specific usage (i.e. at certain times, for specific services, to given destinations and/or from specific sources, with accurate constraints per service/destination). Thus, cSLSs in this case will provide strong guarantees, contractually enforced by an accurate definition of the customers' requirements.

Finally, the following general assumptions/constraints are made in order to build solutions that are adequate and deployable in the Internet.

- Networks should be ready to convey inter-domain QoS traffic before cSLS agreements are negotiated (as is the case with inter-domain routing).
- The MESCAL proposal does not make any assumptions on the applications that will use the QoS capabilities, allowing in for the support of unanticipated applications.
- Whenever a QoS route to destinations is not available, the best effort route may be used as an alternative.

3.2 Customer and Provider Requirements

3.2.1 Introduction

Increasing the deployment of QoS-based services across the Internet requires a large set of providers to cooperate. This cooperation raises a number of complex challenges for Internet operators, not only due to the complexity of the technical issues to be solved but also due to the lack of appropriate standardized contractual agreements and automatic negotiation mechanisms between providers. To this aim, MESCAL will design a suitable inter-domain IP QoS architecture and appropriate solutions.

A first step to achieve the above task is to list the requirements related to the actors involved in the QoS delivery chain. This section aims at identifying requirements from both providers' and customers' perspectives. Therefore, the proposed MESCAL solution will address such requirements.

Then, for evaluating an Internet QoS solution against a specific requirement, the solution will have to:

- Indicate what level of support it offers for each of the requirements: F (Full) means that all implications of the requirement can be fulfilled, M (Medium) means that a significant part of the requirement can be satisfactorily met, L (Low) otherwise.
- Give an explanation of the above rating, i.e., whatever the result of the evaluation; concrete features must be put forward to justify the rating. Especially, when the solution is said to 'Fully' meet the requirement, a detailed justification of all the points tackled by the requirement description should be provided.

The listed requirements are described as follows: For each requirement a general definition is given. Then, a description of its applicability in the Inter-domain QoS delivery context is provided, by outlining the extent at which the requirement will be considered by the project.

3.2.2 Provider requirements

3.2.2.1 Introduction

This section presents the set of provider requirements that MESCAL should address to ensure end-to-end QoS delivery. The purpose is to give an exhaustive and precise definition of requirements against which different solutions will be judged and their applicability evaluated.

3.2.2.2 Description of requirements

3.2.2.2.1 P1: Extend the geographical scope of its QoS services

Definition: The ability for a provider to furnish a level of inter-domain QoS equivalent to the one it can offer to its customers for intra-domain traffic.

Applicability to Inter-domain QoS delivery context:

The MESCAL solution should ensure that a provider is enabled to have at its disposal QoS offers, spanning beyond its domain i.e. across multiple ASes, with the levels of QoS being coherent with the ones it is able to offer to its customers for intra-domain traffic.

More specifically, the intent is to enable a provider to extend its QoS classes (notion of e-QC) over multiple domains, apart from its own, thus enabling the provider to offer reachability to networks beyond its own domain with QoS parameters similar/close to what it could provide within its own domain.

This requirement breaks down into the following two non-exclusive cases, regarding how the expandability of the QoS span of a provider is meant:

- Limited expandability: The provider is able to offer QoS reachability only to specific networks outside its domain. In this case, different QoS levels may apply to different networks. That is, a particular QoS level may only be experienced when reaching a specific destination network.
- Unlimited expandability: The provider is able to offer QoS reachability to (almost) any destination in the Internet, much like as today reachability is offered in the Internet at best-effort QoS levels. The offered QoS levels apply to all destinations.

The above cases are distinguished because they refer to different business models and because they may require different technical solutions e.g. in the first case it may be better to build inter-domain VPNs (e.g. MPLS-based), whereas in the latter case it may be better to build a QoS-aware IP layer across the Internet.

Obviously, the above cases depend on corresponding capabilities of other providers. As such, the requirement of expanding the geographical scope of QoS services in a provider domain entails the following sub-requirement: What are the QoS reachability capabilities assumed to exist in the other providers? The MESCAL solution options should clearly identify the QoS reachability capabilities assumed by the other provider domains.

3.2.2.2.2 P2: Find QoS partners quickly and easily

Definition: The ability to easily and quickly determine the appropriate partners (from a business perspective) for expanding the scope of QoS services i.e. with which to establish pSLSs and the way to achieve that.

Applicability to Inter-domain QoS delivery context:

There are two aspects contained in this requirement.

- Offered QoS Class discovery: Solutions should provide appropriate means to enable providers to discover feasible Offered QoS Classes.

- pSLS negotiation: Once a path and QoS values to reach a destination are chosen by a provider, the means to set up the required pSLS(s) should be rapid and easy. This means that the process for establishing pSLSs should be feasible in the sense that it should follow accepted business practices, well-defined, involving finite steps and based on commonly understood notions. Relevant automated means are also desired for speeding-up the process. A provider may need to set up pSLSs with direct peers, or with a remote AS, or pSLSs may need to be established between two remote ASs upon request from a third-party AS. In the two latter cases, the information necessary can be provided by the means used for QoS path discovery, as described above.

3.2.2.2.3 P3: Verify the fulfilment of the contract

Definition: The ability to check that what is provided conforms to what has been stated contractually.

Applicability to Inter-domain QoS delivery context:

The solution must enable conformance verification of the actual service against the contractual expectations. This should be true for both cSLSs and pSLSs. In either case, the networks' configurations and policies derived by the MESCAL system must ensure that the QoS parameters negotiated in the contract are respected. Some tools or monitoring points must be available to check the conformance of the measured QoS service towards what has been negotiated.

Related to the above, the solution must state relevant tools and information, which are assumed to be provided by other providers.

3.2.2.2.4 P4: Accounting, charging and billing

Definitions:

- Accounting: Technical process of collecting usage records from network nodes such as sender, receiver or router.
- Charging: Transforming the usage records into monetary units and associating them with the user's identity.
- Billing: Collecting charging records, summarizing their charging content, and delivering a bill to a customer including an optional list of detailed charges per user, per service.

Applicability to Inter-domain QoS delivery context:

Not considered by MESCAL.

3.2.2.2.5 P5: Scalability

Definition: Ability for the system to function effectively and keep its performance in desired levels, as the size of the parameters influencing its behaviour increase. In other words, the proposed MESCAL solution should be able to keep its performances unaffected whatever the size of domain span, which could be expressed in terms of number of participating domains (and routers), whatever the number of (c/p)SLSs to be dynamically negotiated and invoked. Performances of the system should also be kept unchanged whatever the volume of the QoS-related information that will be propagated across domains, and without affecting the overall stability and (access) availability of the IP networks themselves.

Applicability to Inter-domain QoS delivery context:

The scalability of the MESCAL solution should be evaluated. This entails the assessment of the complexity of the decision-making components.

Typical size parameters to take into account include:

- Per AS: average number of peers, average number of QCs
- Globally: number of participant ASs, number of required/established pSLSs, number of e-QCs, and number of cSLSs.

3.2.2.2.6 P6: Manageability

Definition: Ability for the system to be managed easily.

Applicability to Inter-domain QoS delivery context:

There are two main domains covered, which must be tackled by MESCAL, in this area:

- Configuration
  - The base configuration, which is intrinsic to the solution, must be tolerable and automation must be provided.
  - The configuration induced by the enforcement of a newly agreed pSLS must not be too heavy, nor make the system unstable (even briefly).
  - The impact of an external modification (for instance, a modification of an intra-domain QC) must be limited, and must not leave the system unstable (even briefly).
- Monitoring
  - The system must offer specific points of visibility for monitoring and feedback purposes (different from the traditional ones, SNMP MIBs, COPS PIBs…etc)

3.2.2.2.7 P7: Resiliency

Definition: Ability for the system to recover from a failure by repairing itself automatically without having to restart the service.

Applicability to Inter-domain QoS delivery context:

Within MESCAL, this means among others that, in case of failure (e.g. link rupture, router breakdown), the system must be able to find/propose another path of equivalent QoS for the impacted destinations. This operation must ensure that all active flows are automatically redirected correctly (e.g. no routing loops) with a minimum of disruption. Notably, a renegotiation of the cSLS conditions former to the failure must be avoided, the system being responsible for providing a satisfactory alternative.

One particular aspect concerning resiliency is security. The following questions should be addressed:

- Does the system present points, which could be exploited by hackers? Are there well-known possible points of failure, whose malfunction could lead to an unavailability of the system?

3.2.2.2.8 P8: pSLS management flexibility

Definition: Degree of freedom for an AS to modify its pSLSs.

Applicability to Inter-domain QoS delivery context:

pSLSs should be viewed as managed entities. As such, providers should be given means for requesting, establishing, modifying and deleting pSLSs. Caution should be taken to ensure that the modification of pSLSs do not disturb, but is in accordance with the requirements of other pSLSs relying on the pSLSs under modification.

In case of pSLS deletion, means must be provided to ensure the coherence and stability of the system, notably the good handling and management of pSLSs that were relying on the deleted pSLS (in a cascading approach). Possible solutions are for instance: forbid the deletion, notification to peers so that they modify their pSLSs before deletion is completed…

3.2.2.2.9 P9: Deployment easiness

Definition: How long and difficult it would take to have all the building blocks ready for operation, that is to say, to begin actual inter-domain communications with QoS activated.

Applicability to Inter-domain QoS delivery context:

Easiness of deployment depends on a number of parameters, such as: number of new protocols required, degree of adherence of the proposed solutions to the market and capabilities of commercially available routers, magnitude of required modifications to existing protocols, impact on intra-domain routing, impact on inter-domain routing and required conformance of other providers with the proposed solutions. The MESCAL solution(s) should clearly identify and describe such aspects.

3.2.2.2.10 P10: Backward compatibility

Definition: The risk and impact on the infrastructure already in place, when deploying the MESCAL solutions.

Applicability to Inter-domain QoS delivery context:

In order to achieve the goals pursued by MESCAL, proposed solutions are likely to introduce more or less modifications on the existing infrastructures. The MESCAL approach should provide the adequate guarantees as far as the backward compatibility issue is concerned, not only for allowing a smooth migration, but also to prevent existing infrastructures from being unusable and instable.

Among other criteria, the following are considered as important to judge the fulfilment of this requirement:

- The impact on the intra-domain routing process must be as limited as possible.
- The impact on the inter-domain routing process must be as limited as possible.
- When in operation, the MESCAL system must not introduce instability neither on the network itself, nor on the already deployed and running services.

3.2.2.2.11 P11: Applicability to business model

Definition: To what business case(s) the solution is applicable.

Applicability to Inter-domain QoS delivery context:

As there are different business cases in offering Internet QoS services, the MESCAL solutions should be clearly positioned as to which type of business cases they can address.

Different business cases can be seen along the following views:

- Customer view:

  A typical mass-market customer, is potentially interested in accessing any kind of service in any location in the Internet and at any time.

  A typical enterprise customer, is focussed on a well- known and limited set of services whose location, duration, QoS constraints, can be perfectly defined.

- Provider view:

  The provider wishes to extend its own QoS services to external users at specific or any network in the Internet.

3.2.2.2.12 P12: Multicast aspects

Definition: Support for delivering multicast-based IP services in the Internet.

Applicability to Inter-domain QoS delivery context:

It is important to evaluate the impact of supporting multicast-based services on the features and performance of the approach along the following lines:

- Does the multicast support imply major changes or add-ons to the unicast model?
- Does the multicast service address all aspects (and customers) listed in the business model
- How to manage the replicated multicast traffic within the network?

- How to avoid imposing significant impacts on the underlying IGMP, PIM-SM, MBGP protocols, as well as core router architecture for including DiffServ aware multicast services?
- How to handle the scalability issues concerning QoS deployment?
  - Low overhead for group/QoS state maintenance within core networks.
  - No traffic conditioning capability within DiffServ core routers.

3.2.3 Customer Requirements

3.2.3.1 Introduction

This section presents requirements from the perspectives of the customers of QoS-based Internet services. The requirements are drawn from current business practices and market needs as understood by the project partners. The requirements pose corresponding requirements to the providers of QoS-based Internet services, which in turn need to be taken into account by the solution proposed by MESCAL.

Considering a provider, the term "customer" is taken to denote either an end-customer (recipient of QoS services), or another provider. Unless explicitly stated to denote a particular type of customer, the term "customer" is used to denote either of these types of customers.

3.2.3.2 Customer Requirements details

3.2.3.2.1 C1: Characteristics of QoS Services

Definition: Ability of customers to send/receive traffic with end-to-end QoS guarantees to/from destinations in the Internet.

Applicability to Inter-domain QoS delivery context:

This general requirement can break down into the following requirements:

- On the topological scope of the services: Customers should be able to send/receive traffic to/from specific and/or any destination in the Internet. That is, given the sites of a particular customer, the customer should be able to:
  - Send traffic with end-to-end QoS guarantees to specific destinations i.e. only to destinations, which have been a-priori agreed with the provider.
  - Send traffic with end-to-end QoS guarantees to any possible destination; of course, at the time of actually requesting the service, the destination should be clearly specified in the IP address space.
  - Receive traffic with end-to-end QoS guarantees from specific sources.
  - Receive traffic with end-to-end QoS guarantees from any possible sources.
- On the QoS: The QoS guarantees should refer to well-defined performance metrics reflecting the quality of the service from the customer's perspective. At the network layer, these metrics should reflect the packet transfer quality e.g. throughput, one-way transit delay, inter-packet delay variation, and packet loss. Note that, since MESCAL is concerned with connectivity QoS-based services only, these network-level metrics also make sense from customer perspectives. The end-to-end QoS guarantees should be clearly specified, commonly understood and mutually agreed by the customers and the providers. Related to this requirement are the following requirements:
  - The QoS could be quantitatively specified e.g. by means on certain bounds on related performance metrics.
  - The QoS could be qualitatively specified e.g. relatively to other QoS levels by means of appropriate qualifications such as golden, silver, bronze QoS levels.

- Customers should be able to freely choose their QoS-based services according to their actual needs. Customers should be ideally offered with a choice of QoS-services, even similar services at different QoS levels. However, when the service is actually requested, its QoS levels should be clearly and unambiguously defined.

The above requirements are distinguished because they refer to different types of customers, in terms of their requirements in using QoS services; therefore corresponding to different business cases. Some customers may know in advance the type of QoS services they require, whereas some others may not.

From a provider's perspective the above requirements yield the following requirements:

- The SLSs (pSLSs or cSLSs) underlying the offering of QoS-based services should be able to:
  - Capture the QoS characteristics of both upstream and downstream traffic (with respect the premises of a customer),
  - Specify the QoS characteristics quantitatively and/or qualitatively, and
  - Leave appropriate degrees of freedom in specifying the destinations and/or the QoS-levels of the QoS services, as required for covering the diverse needs of the customers, obviously according to the service provisioning capabilities of the provider.
- To be able to expand the geographical span of the offered QoS services beyond the provider domain –refer to corresponding provider requirement P1 in section 3.2.2.2.1.

3.2.3.2.2 C2: Dynamic Service Subscription

Definition: Ability of customers to dynamically subscribe and unsubscribe to QoS services, as per their communication needs.

Applicability to Inter-domain QoS delivery context:

Subscriptions should not be taken for granted as long-lived service contracts. Subscriptions may well be short-lived e.g. for a weekend. In fact, given the multi-service, multi-provider nature of the telecommunications market and the dynamic nature of customer needs –not all customers may know in advance their QoS service needs-, the ability to establish SLSs is a key aspect of service offering. Some customers may be more attracted by such dynamic service offerings compared to static, monolithic offerings, as their service needs continuously evolve.

From a provider's perspective, this requirement yields the following requirements:

- Providers should provide means for enabling customers to modify and terminate existing service contracts (SLSs).
- Providers should provide means for enabling customers to subscribe to QoS services on-demand and for short time periods, upon customers' requests.

Automated means for enabling subscription e.g. through the Web and for handling subscription requests e.g. service configuration/activation means, could facilitate the satisfactory fulfilment of the above requirements.

3.2.3.2.3 C3: Service Invocation

Definition: Ability of customers to invoke i.e. to actually request QoS services. Services are invoked by the users, within the subscription profiles (as described in the SLSs) agreed between the customer and the provider.

Applicability to Inter-domain QoS delivery context:

This requirement entails the following:

- Customers should be able to invoke the services either explicitly or implicitly. Explicit invocation will probably yield the use of an explicit QoS signalling protocol. Implicit invocation does not

require the explicit use of a QoS signalling protocol; users can initiate their flows at any time, as long as the corresponding streams adhere to agreed subscribed profile.

- Customers should be provided with appropriate means to invoke their QoS services. These means should be in accordance with the QoS service specifications i.e. should be able to convey the required information for identifying the particular QoS service requested, as specified by MESCAL.

From provider perspectives, this requirement yields the following requirements:

- Providers should be able to support both explicit and implicit service invocations. As for the former case is concerned, providers should be able to support the termination and handling of appropriate QoS signalling protocols.
- In either case, the invocation means should be capable of conveying the MESCAL QoS SLSs; either as part of the QoS signalling protocol used or through the information included in the IP header. The conveyed information should help providers in unambiguously identifying the MESCAL-conforming requested QoS service and the customer requesting it.
- Providers should provide for automated means in authenticating and authorising a (implicitly or explicitly) request of a QoS-based service.

3.2.3.2.4 C4: Verify the fulfilment of the contract

Definition: Ability of customers to assess on-line that the invoked services are provided in accordance to the agreed QoS levels.

Applicability to Inter-domain QoS delivery context:

Customers should be able to check that the quality of the services they have subscribed to is in accordance with what they have agreed with the provider. This requires that they should be provided with appropriate self-monitoring tools.

From a provider's perspective, this requirement yields the following requirements:

- Providers should provide customers with appropriate monitoring tools, enabling the customers to assess the QoS of the services they request.
- Providers should cater for appropriate means for receiving and analysing customer complaints with respect to the received services.

3.2.3.2.5 C5: Multicast Aspects

Definition: Ability of customers to initiate and/or participate to multicast groups with some QoS guarantees.

Applicability to Inter-domain QoS delivery context:

Since almost all the multicast services are receiver oriented, the following is from the perspectives of multicast receivers (group members):

- Receivers desire to receive specific multicast traffic from the subscribed group, and hence the functionality of source filtering is needed to avoid delivering unwanted multicast traffic.
- Receivers should be able to specify their QoS requirements individually, i.e. different recipients could specify different QoS levels via receiver-oriented cSLSs for multicast traffic.

4 THE MESCAL QOS SERVICE MODEL (DEFINITIONS)

4.1 Introduction

Current business practices prove that there is not (cannot be) a single provider offering global coverage of the whole Internet. As such, providers need to interact between them so as to expand the geographical scope of the services they offer. Considering QoS-based services, these interactions may not exclusively occur at the network (IP) layer; they may also occur at the service layer on the basis of specific service agreements.

In the above scenery, this document introduces the MESCAL Internet service model, which aims at laying down the notions, entities and relationships between them, pertinent to the issue of definition and provisioning of QoS-based services in the Internet, across multiple Provider domains. In other words, the MESCAL Internet service model presents the informational architecture/the basic 'service vocabulary' for building/defining Internet QoS-based services.

From another angle, the MESCAL service model outlines the requirements of Internet QoS-based services from an informational viewpoint. As such, it sets the functional targets of the service offering and provisioning functionality, while it also presents the necessary abstractions in the service layer around which this functionality needs to be designed.

The MESCAL model relies on the QoS service model proposed by TEQUILA [TEQUI], [Trimin03], for QoS-based intra-domain services. The MESCAL model extends the TEQUILA model to cover QoS-based services spanning the whole Internet, rather than a domain of a particular provider.

4.2 Notions and Entities

This section presents the notions and entities of the MESCAL Internet QoS model.

4.2.1 QoS-based Services

4.2.1.1 Definitions

The term *service* denotes, from customer perspectives, a specific offering made by a provider, which (offering) should clearly and unambiguously describe what it offers and the terms and conditions under which it could be used. Equivalently, from provider perspectives, a service denotes a subset of the provider's domain capabilities with a clear description of the what's and how's regarding its use by customers or third parties in general.

The term *QoS-based service, or just QoS service* denotes a service that is believed to entail a sort of added value to customers e.g. matching application and customer usage requirements.

The current trend in service offering is contract-based. Services are offered on the basis of the so-called *Service Level Agreements (SLAs).* SLAs are established between customers and providers and describe the characteristics of the service and the mutual responsibilities of each party (customer, provider) for using/providing the service. In SLA-based service offering then, on one hand services should be described comprehensively enough so that can be understood by the customers, on the other hand providers should ensure that the characteristics of the services, as depicted in the SLAs, are indeed provided as agreed. SLAs may also be established between two providers -with one provider acting in a customer role and the other in a customer role- to back-up agreements at service level for expanding the geographical span of their services (see also section 4.2.1.2). SLAs between providers extend the notion of peering business agreements that exist today between providers for mutually exchanging traffic at given rates, or even without any financial settlement [Huston99]. Obviously, in a QoS-based Internet such agreements do not present a viable model; they need to include description of service characteristics, accounting and billing aspects, hence the need for SLAs.

The term *Service Level Specifications (SLSs)* denotes the technical characteristics of a given service in the context of an SLA. The technical characteristics of a service refer to the network level

provisioning aspects of the service e.g. request, activation and delivery aspects from network perspectives. Non-technical service provisioning aspects such as billing and payment aspects, are not part of SLSs; they are part of the overall SLA. SLSs are integral part of SLAs, and conversely SLAs include SLSs.

MESCAL is concerned with SLSs. Service accounting and billing aspects are outside the scope of MESCAL investigation.

4.2.1.2 On SLSs – cSLSs and pSLSs

MESCAL distinguishes two types of SLSs (and subsequently SLAs): *cSLSs* established between customers and providers and *pSLSs* established between providers.

The providers between which pSLSs are established may not necessarily be interconnected. In the general case, a provider (acting in a customer role) may establish pSLSs with a remote provider (acting in a provider role), should the latter be appropriately located and contacted.

The term *peering providers* is used to denote providers, which are interconnected; and, the term *service-peering providers* is used to denote providers between which pSLSs have been established.

The following operations on SLSs (and SLAs) should be allowed: *establishment of new SLSs, modification and termination of already established SLSs*. To this end, appropriate means should be provided, including informational models for describing SLSs and well defined manual and/or automated procedures for discovering, requesting and agreeing on the establishment, modification and termination of SLSs. Such procedures should provide for negotiation semantics/primitives for overcoming the limitations of a monolithic 'yes/no' type of interaction. The Service Negotiation Protocol (SrNP) specified by TEQUILA [TEQUI] is an example of such automated negotiation means.

Two styles in requesting and subsequently establishing SLSs can be distinguished: *restricted SLS request style* and *unrestricted SLS request style.* Under the restricted SLS request style, a requestor (customer or provider acting in a customer role) requests from a provider the establishment of SLSs, which refer only to currently offered services. Under the unrestricted request style, a requestor may request from a provider the establishment of SLSs referring to services that may need additional capabilities than the ones provided by the currently offered services. In a sense, the unrestricted request style is equivalent to the restricted request style with an addition of the nature 'please send any other request to the marketing department'.

The above differentiation is necessary for capturing different business strategies instigating the establishment of SLSs as well as decisions regarding the services to be offered. For instance, providers could allow for an unrestricted style, as a means to 'grasp' needs for future services. Furthermore, this differentiation is helpful for deriving requirements for negotiation procedures and associated logic.

All the above aspects on SLSs and their establishment, are deemed essential in the arena of service provisioning in the Internet, where in addition to advances in the network (IP) layer, appropriate 'hooks' for capturing business level objectives and policies need to be catered for.

4.2.1.3 MESCAL Service Focus - Connectivity Services

MESCAL is concerned with *QoS-based connectivity services.* A connectivity service is a 'get-through'/ 'traverse' service for reaching particular destination(s) from specific source(s) in the IP address space. The QoS aspects of connectivity services mainly refer to the quality at which the user-transmitted IP datagrams are transferred by the network between user-ends. Higher-level, informational, application-specific services e.g. streaming or video-on-demand services are outside of the scope of MESCAL. Note, that the latter services usually have a connectivity dimension, which if not provisioned properly, would lead the whole service not be provisioned at all. Therefore, connectivity services should be studied first, before moving to higher-level services.

MESCAL distinguishes QoS-based connectivity services into *elementary and complex connectivity services*. Elementary connectivity services are strictly point-to-point and unidirectional, whereas complex connectivity services may be multi-point-to-multi-point and bi-directional. As such, complex

connectivity services encompass a number of elementary connectivity services as appropriate to the context of the connectivity service itself; equivalently, elementary connectivity services can be viewed as the 'connectivity legs' (the 'nucleus') of complex connectivity services. Typical examples of (complex) connectivity services include VPN, Internet access, server access services. Complex connectivity services constitute the connectivity services actually offered to the customers, whereas elementary connectivity services can only exist in the context of these services and as such are not offered to customers. As such, the term connectivity service used throughout this document implies a complex connectivity service.

The distinction between complex and elementary connectivity services is deemed helpful for decomposing the provisioning of connectivity services from the perspectives of a provider. Furthermore, this distinction may be used to facilitate the specification of SLSs. As a complex connectivity service is comprised by a number of elementary connectivity services, its SLS may be comprised by the SLSs of its constituent elementary services; therefore, SLSs may only be specified for elementary connectivity services. In line with this view, TEQUILA has specified its SLS template for intra-domain QoS-based connectivity services [Goder02].

Figure 4 depicts the QoS-based service hierarchy as assumed by MESCAL; from higher-level, informational, application-specific services to complex and elementary connectivity services.

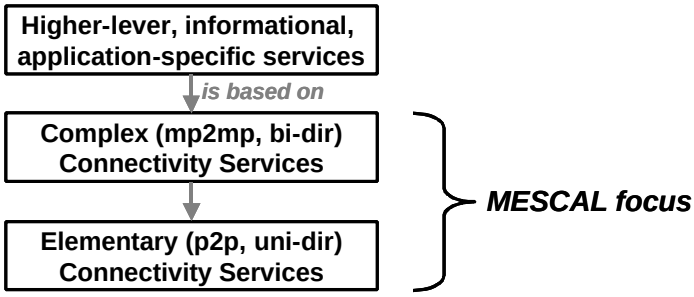


Figure 4: QoS-based service hierarchy and MESCAL focus.

Based on the roles identified in the MESCAL business model, QoS-based connectivity services are offered by the so-called 'IP Network Providers', who own and administer an IP network infrastructure including customer access means. As already said, 'IP Network Providers' need to interact between them so as to expand the geographical scope of the QoS services they offer. These interactions may occur at a network (IP) and/or service layer on the basis of respective pSLSs. The following sections introduce appropriate notions underlying these interactions.

Throughout this document the term service denotes a connectivity service and the term provider an 'IP Network Provider', unless otherwise specified.

4.2.2 QoS-classes

4.2.2.1 Definitions

As outlined in the previous section, QoS-based services reflect and need to be supported by corresponding 'capabilities' of the provider domains across the Internet. As such, the following definitions are put forward:

A *QoS-class (QC)* denotes a basic network-wide *QoS transfer capability* of a Provider domain.

A QoS transfer capability is a set of attribute-value pairs, where the attributes express various packet transfer performance parameters such as one-way transit delay, packet loss and inter-packet delay variation (jitter), and their values have the meaning of upper bounds on them. Considering the statistical nature of the packet transfer performance parameters, the corresponding attributes may not be invariant; rather, they could refer to specific time-intervals, denoting moving averages and/or percentiles or inverse percentiles (confidence levels for being below a given threshold). Furthermore, the attribute values (bounds) may not be absolutely defined; they may be qualitatively defined

relatively to the corresponding values of other QoS-classes. In essence then, a QoS-class is a set of packet *transfer performance parameters* (attributes) associated with specific *performance targets* (values).

It should be noted that QoS-classes are not services per se; their definition does not entail service provisioning semantics and aspects e.g. activation modes, user identification and usage requirements. The concept of QoS-class could be compared to the notion of Per Domain Behaviours (PDBs) – debated in the DiffServ workgroup of the IETF in the recent past.

QoS-classes are associated with a number of constraints, which denote conditions for their time- and topology-wise availability. Time-related constraints are expressed in period(s) of day/week/month during which the QoS-class can be (or cannot be) made available. Topological constraints are expressed in terms of reachable domain boundaries (e.g. IP network prefixes) between which the QoS-class can be (or cannot be) made available.

Considering a provider's domain, the provisioning of a QoS-class may solely rely upon the domain's own network engineering abilities, those related to routing and resource (bandwidth and buffer) management, which result by combining the elementary IP DiffServ QoS capabilities with intelligent traffic engineering functions and related policies. In addition to the domain's own engineering abilities, the provisioning of a QoS-class end-to-end may rely upon the QoS transfer capabilities (QoS-classes) provided by other provider domains, should the latter could be made known and used; hence the necessity of interactions between providers.

We distinguish between *local-QoS-classes* and *extended-QoS-classes*. Namely, given a provider domain:

A l*ocal-QoS-class (l-QC)* denotes a basic network-wide QoS transfer capability that can be provided by means employed in the provider domain itself. Evidently, the domain boundaries appearing in the topological constraints of an l-QC should belong to the boundaries of the provider domain.

An *extended-QoS-class (e-QC)* denotes a basic network-wide QoS transfer capability that can be provided by means employed not only in the provider domain but also utilising appropriate means in other (service-peering) provider domains. In other words, an e-QC is provided by combining the QoS transfer capabilities (QoS-classes) of the provider domain with appropriate capabilities (QoS-classes, l-QC or e-QC) of other provider domains. The domain boundaries appearing in the topological constraints of an e-QC could be outside the boundaries of the provider domain, thus extending the topological scope of the QoS transfer capabilities of the provider domain.

The above distinction is required for capturing the notion of 'QoS capabilities' across domains, upon which QoS-based Internet services are/could be built. In a sense, this distinction is analogous to the intra-/inter-domain distinction that usually applies in the context of the Internet.

Hereafter, the term QoS-class (QC) denotes either a local or an extended QoS-class, unless it is explicitly said to mean a particular one of the two.

4.2.2.2 Comparisons between QoS-classes

By comparing the values of the corresponding QoS-class performance parameters, an ordering relationship could be defined amongst QoS-classes. The following definitions are put forward:

A QoS-class A is said to be *"at least as good as"* a QoS-class B, conversely QoS-class B is said to be *"at most as good as"* a QoS-class A, denoted by *"A ≥ B"* if and only if the values of *all* corresponding performance parameters of the QoS-classes A and B are accordingly ordered. The 'accordingly' qualification refers to the nature of the QoS-class performance parameters (attributes) as discussed in the previous section (moving averages, percentiles etc.). For instance, if the QoS-class attributes denote averages over the same period of time, their values (bounds) should be ordered according to the ≤ relationship, whereas if the QoS-class attributes denote inverse percentiles for a given threshold, their values should be ordered according to the ≥ relationship. Obviously, the attribute values to compare should be expressed in the same or convertible units.

Similarly we define the case that a QoS-class A is *"better"* than a QoS-class B, conversely that QoS-class B is *"worse"* than a QoS-class A, denoted by "A > B".

The above definitions could be extended to define a *lexicographical ordering* between QoS-classes. In this case the performance attributes of the QoS-classes should be appropriately prioritised viewing QoS-classes as ordered vectors of performance parameters; the first co-ordinate reflecting the most significant performance parameter and so on. Then, QoS-classes can be ordered by checking the values of the corresponding attributes per co-ordinate, not by checking the values of all corresponding attributes as in the previous definitions.

It should be noted that the defined *ordering relationship is partial*, not total, meaning that not every pair of QoS-classes can indeed be compared. For instance, this could be the case when the corresponding attributes of the QoS-classes to compare are of not of similar nature e.g. averages over different time periods or averages versus percentiles, making the comparison of their values infeasible. Alternatively, such cases could appear when QoS-classes are not compared lexicographically and *some* of the values of corresponding QoS-class performance parameters are accordingly ordered, whereas *some* others are not.

Because the QoS-class ordering relationship is partial, there might be a number of *"best"* or *"worst"* QoS-classes instead of a single such element, even if the set of QoS-classes is finite.

4.2.2.3 Types of values of QoS-classes

Orthogonal to the interpretation and the nature of the QoS-class performance parameters (attributes) as discussed in section 4.2.2.1, the values -and subsequently the QoS-classes- may be distinguished into different types according to how these values are assumed. The values may be nominal or actual. Nominal values are set/deduced theoretically, whereas actual values are set/deduced from operational practices. Both nominal and actual categories of values are subject to the specific business policies and operational practices of the particular Provider administration regarding service provisioning. Table 1 presents possible types of nominal and actual values.

QoS-class parameter values		
Category	**Type**	**Description**
Nominal – Set	Targeted	Values set as objectives for engineering the network, setting the targets of the off-line traffic engineering functions that dimension the network. These values are deduced by the requirements of widely deployed applications (cf. the notions of Meta-QoS-Class and global-QoS-class below) and/or market needs.
Nominal – Deduced	Engineered	Values yielded as a result of the off-line traffic engineering algorithms run to dimension the network so as to be able offer QoS-classes at their 'targeted' values. These values take into account the characteristics of the physical network configuration and topology, and their validity is subject to the errors inherent in the mathematical models used. These values should be as least as good as the corresponding 'targeted' type values.
Actual – Set	Offered	Values as assigned by the actual service offering activities i.e. values deemed appropriate for creating competitive service offerings to third parties (customers or providers). That is, these values are exported in the SLSs. Considering that QoS-based services should be in accordance with the capabilities of the domain, these values should be primarily at least as good as what is deemed 'attractive' to customers, while close to the corresponding 'engineered' or 'targeted' type values. These values may change as the corresponding policies for service offering change. They may be assigned either in absolute terms or qualitatively, relatively to the corresponding values of other QoS-classes.
Actual – Deduced	Measured	Values yielded by actual measurement during network operation. These values may be in any relation with the previous types of values. Ideally, they should be –on average- over a sufficiently large timescale, less than the corresponding engineered types of values and should not violate (at all) the 'offered' values. They may be used for validating and/or advertising the performance of the network. These values change as network traffic conditions change.

Table 1: QoS-class parameter value types.

In the above cases where the QoS-class parameters values (bounds) can be set and not deduced (i.e. 'targeted' and 'offered' type cases), the determination of appropriate values is subject to relevant business policies regarding service provisioning, taking into account requirements of well-known applications/services (cf. the notions of 'Meta-QoS-Class' and 'global-QoS-class' below), perceived

user needs and current/emerging market trends. The deduced QoS-class parameter values (i.e. 'engineered' and 'measured' type cases) are influenced by policies at network operation level. E.g. 'engineered' values may be influenced by policies determining the desired network-wide load balancing levels and 'measured' values are subject to the policy-set measurement parameters.

Obviously, the number of QoS-classes supported by a provider domain corresponds to the number of distinct values, which are actually set to the QoS-class performance parameters.

Based on the identified QoS-class parameter value types (cf. Table 1), the following terminology is introduced:

Targeted-QoS-class (t-QC), engineered-QoS-class (eng-QC), offered-QoS-class (o-QC), measured-QoS-class (m-QC) denotes a QoS-class where the values of its performance parameters are of 'targeted', 'engineered', 'offered', 'measured' type, correspondingly. The following statements are true regarding the relationship of these QoS-class types:

- By definition, there should be: o-QC ≤ t-QC ≤ eng-QC
- While eng-QC ≤ m-QC, the traffic-related objectives of traffic engineering are satisfied.
- When t-QC ≤ m-QC < eng-QC, re-engineering of (parts of) the network has to be considered.
- When o-QC ≤ m-QC < t-QC, the network must be re-engineered urgently.
- When m-QC < o-QC, the service contracts cannot be fulfilled anymore and the network must be re-engineered or even additional resources to be brought in.

4.2.2.4 Offering and Using QoS-classes

As QoS-classes reflect capabilities, this section addresses the question of 'what can these capabilities be used for?' or equivalently, 'how can these capabilities be used by third parties?'

Considering a Provider domain, QoS-classes may be used in either (not exclusively) of the following two cases:

- *For offering QoS-based services* to customers or other providers. In this case, the values of the QoS-classes may be of 'offered' or 'targeted' types (cf. Table 1).

  QoS-classes are building blocks for offering and provisioning QoS-based connectivity services – not the services themselves. Conversely, QoS-based connectivity services should be mapped to QoS-classes. In essence, from the perspectives of service offering, QoS-classes express the transfer quality aspects of the QoS-based connectivity services; and, from the perspectives of service provisioning, QoS-classes segregate the network QoS-space into a number of distinct classes, aggregating user QoS traffic accordingly. In this respect, the notion of QoS-classes sets the traffic-related objectives of the traffic engineering functions, prompting for approaches such as the ones following the Bandwidth Constraints model in the context of DiffServ-aware MPLS traffic engineering [Lefau03a] -the Russian Dolls Model [Lefau03b], the Maximum Allocated bandwidth Model [Lefau03c]- or the TEQUILA 'initially plan then take care' approach [Trimin01]. It should be stressed that the notion of QoS-classes does not necessarily prompt for hard bandwidth reservations per QoS-class in the network, as for instance in the TEQUILA approach.

- *For 'pure' informational purposes* that is, for announcing the QoS transfer capabilities of the provider domain. In this case, QoS-classes are announced 'as is' i.e. without service semantics. The values of the QoS-classes may be of 'targeted' or 'offered' or 'measured' types. Announcements could be done through various means, protocol- and/or platform-based, either periodically or asynchronously based on well-defined triggering conditions.

  Capability announcements are mainly targeted at service-peering providers, since QoS-classes do not bear service semantics, which are of interest to customers. They could also be targeted at customers, being provided as part of an agreed service. Providers might find useful to announce their QoS-classes –QoS transfer capabilities- for attracting service-peering providers for the purpose of increasing their revenue-earning sources (volumes of terminating and transiting QoS

traffic), furthermore for expanding the reach of the supported QoS-class capabilities on a mutual basis.

The substantial difference between the above cases lies in the implications incurred for the provider domain. In the first case, the provider is formally obliged to honour the terms and conditions underlying the offering of their services (SLS/SLAs). In the second case, the provider does not assume such formal obligations, as it (the provider) is not bound to any agreement, though it needs to uphold its announcements for the sake of its integrity and reputation.

Conversely, considering the cases above, QoS-classes supported by a provider domain can be used by other providers or customers in either of the following two cases:

- *Contentedly*, through corresponding QoS-based services. In this case, the use of QoS-class capabilities is done implicitly (indirectly) and is bound to mutual agreements underlying service offerings (cf. pSLSs, section 4.2.1). As such, QoS-class capabilities may be used with the guarantees underlying the offering of the corresponding service (cf. section 4.2.2.5).
- *Non-contentedly*, following related capability announcements. As long as a provider domain announces QoS-class capabilities, other provider domains or customers can use directly these capabilities i.e. not through the establishment of SLS/SLAs. In this case, the use of announced QoS-class capabilities is not bound to any agreement and it is on a 'to-do-my-best' basis.

The following point is worth discussing. The 'non-contentedly' use case does not necessarily imply that providers offer their QoS network resources for free. This kind of use case may happen on the basis of mutual business agreements between providers for exchanging aggregate traffic, as they exist today. The 'contentedly' use case extends these 'aggregate traffic exchange agreements', to agreements regarding the exchange/usage of traffic at certain QoS characteristics; these agreements are substantiated in corresponding pSLSs/pSLAs.

The 'non-contentedly' use case can be seen as a special case of the 'contentedly' use case when the services guarantees (cf. section 4.2.2.5), as depicted in the SLS/SLAs, are very loose –even non-existent. As such, without loss of generality *it is considered that QoS-classes can only be used in the context of QoS-based services i.e. in the context of SLSs/SLAs, which may or may not bear service guarantees*.

4.2.2.5 QoS-based Service Guarantees and QoS-classes

When applied to an offered service, the term *QoS-based service guarantees*, or *QoS service guarantees* for short, denotes the guarantees with which the quality aspects of the offered service can be provided from provider perspectives. These quality aspects differentiate similar services amongst them.

Considering QoS-based connectivity services, the focus of MESCAL, we view that QoS service guarantees consist of the following parts:

- *Performance guarantees*, which reflect the quality of the transfer of the user-transmitted datagrams in the context of the service. Considering that QoS-classes are the building blocks of QoS-based services (cf. discussion in previous section), these guarantees directly correspond to the values of (bounds on) the performance parameters of the QoS-class(es), which the offered service is based on.
- *Bandwidth guarantees*, expressed as an upper limit, in bandwidth units, on the user traffic injected in the network up to which the agreed service performance guarantees can be given.
- *Grade of service* denoting the probability of getting through the network valid (according to subscription profile) service requests.

The above types of QoS service guarantees should be reflected in the c/pSLSs, underlying the offering of QoS-based services. It is the responsibility of the provider offering the services to ensure that the above guarantees can be gracefully provided -not significantly violated.

The above classification of QoS service guarantees is in accordance to the view of the 'ippm' workgroup of the IETF, which does not consider bandwidth as a performance parameter. Furthermore, it is in line with the template proposed by TEQUILA [Goder02] for describing SLSs for QoS-based connectivity services.

It should be noted that the definition of QoS-classes prompts for hard or statistical/probabilistic QoS service performance guarantees, depending on the nature of the QoS-class performance parameters (attributes); as already outlined (cf. section 4.2.2.1), these attributes may be invariant, or they could be of statistical nature e.g. percentiles.

4.2.2.6 Provisioning of QoS-classes

It should be noted that the extent (confidence) at which the QoS-classes can be gracefully provisioned i.e. their performance targets –upper bounds on their performance parameters- can be safely met is not considered part of the definition of the QoS-class itself. This aspect entails service semantics (cf. previous section), which are not assumed by QoS-classes.

Therefore, the issue of being able to gracefully provision QoS-classes should be seen only in connection to the way QoS-classes are made available by the provider domain for use, as outlined in section 4.2.2.4). If QoS-classes are used for offering QoS-based services, QoS-class performance targets should be sufficiently met so that service performance guarantees (as specified in the SLSs) are not violated. If QoS-classes are used for announcing domain's capabilities, QoS-class performance targets should be met to the extent deemed necessary for the announcements to be valid.

The provisioning of QoS-classes to the extent desired falls into the realm of the domain's QoS delivery capabilities, combining the DiffServ elementary (nodal) QoS-enabling mechanisms with intelligent traffic engineering functions for QoS-based routing and resource management. In addition, in the case of extended-QoS-classes, QoS-class provisioning is also dependent on the corresponding capabilities of service-peering provider domains, which in turn are dependent on the corresponding capabilities of their service-peering domains and so on. Obviously, the existence of pSLSs between provider domains increases the confidence at which extended-QoS-classes could be provisioned in each domain.

For feasibility, manageability and scalability reasons, the QoS-classes should be pre-determined and fairly restricted in number; otherwise, the likelihood of not being able to manage effectively their provision would prohibitively increase. The fact that the values (bounds) of the performance parameters of the QoS-classes may be set in accordance to known application/service requirements (see following sections) contributes to this direction.

4.2.3 Meta-QoS-Classes

Although there is a plethora of existing and emerging applications and value-added services, their end-to-end transfer requirements do not analogously diverge. Transfer requirements are expressed by desired value-ranges of specific performance parameters -delay, loss, jitter; and the number of such desired ranges is rather limited. As such, the following definition is put forward.

A *Meta-QoS-Class (meta-QC)* denotes an abstract QoS-class, where the 'meta' qualification refers to the range of values of the QoS-class performance parameters (e.g. delay, loss, litter) rather than on the QoS-class parameters (information) themselves. Meta-QoS-Classes describe in a quantitative or qualitative way ranges of QoS-class performance parameter values, rather than particular values as QoS-classes do.

In the case of qualitatively defined Meta-QoS-Classes, performance parameters' value ranges are described by means of the following qualifications: 'very low', 'low', 'any'; note that the values have the meaning of upper bounds. Stringent-QoS-class with delay value='very low' and loss value='very low', or delay-sensitive-QoS-class with delay value='low' and loss value='any' are typical examples of Meta-QoS-Classes.

QoS-classes could be seen as instances of corresponding Meta-QoS-Classes. For a given Meta-QoS-Class, a number of QoS-classes could be defined to adhere to the Meta-QoS-Class definition. Based

on the QoS-class ordering relationship defined in section 4.2.2.2, Meta-QoS-Classes can be arranged hierarchically. Thus, a given QoS-class could adhere to more than one Meta-QoS-Class.

The notion of Meta-QoS-Class may be useful for determining the values of QoS-classes as well as for providing the grounds for 'grouping' (mapping between, see below) QoS-classes of different provider domains.

4.2.4 Global-QoS-Classes

Global-QoS-classes (g-QCs) are QoS-classes, where the values of the performance parameters (considered of 'offered' type, cf. Table 1) express the desired transfer requirements of widely deployed (globally known) services/applications. Typical examples of global-QoS-classes could be VoIP-QoS-class or High-Quality-Video-QoS-class.

For a given widely deployed service/application, a number of corresponding global-QoS-classes could be defined, depending on the nature (e.g. average, percentile) of the QoS-class attributes expressing transfer performance parameters.

Based on the QoS-class ordering relationship defined in section 4.2.2.2, global-QoS-classes can be arranged hierarchically, and multiple QoS-classes could adhere to a specific global-QoS-class.

Similar to the notion of Meta-QoS-Class, the notion of global-QoS-class may be useful for determining the values of QoS-classes as well as for providing the grounds for 'grouping' (mapping between, see below) QoS-classes of different Provider domains.

4.3 The MESCAL Internet QoS Service Model

Summarising the concepts and the notions presented in the previous section, the MESCAL model for Internet QoS-based services is shown in Figure 5.

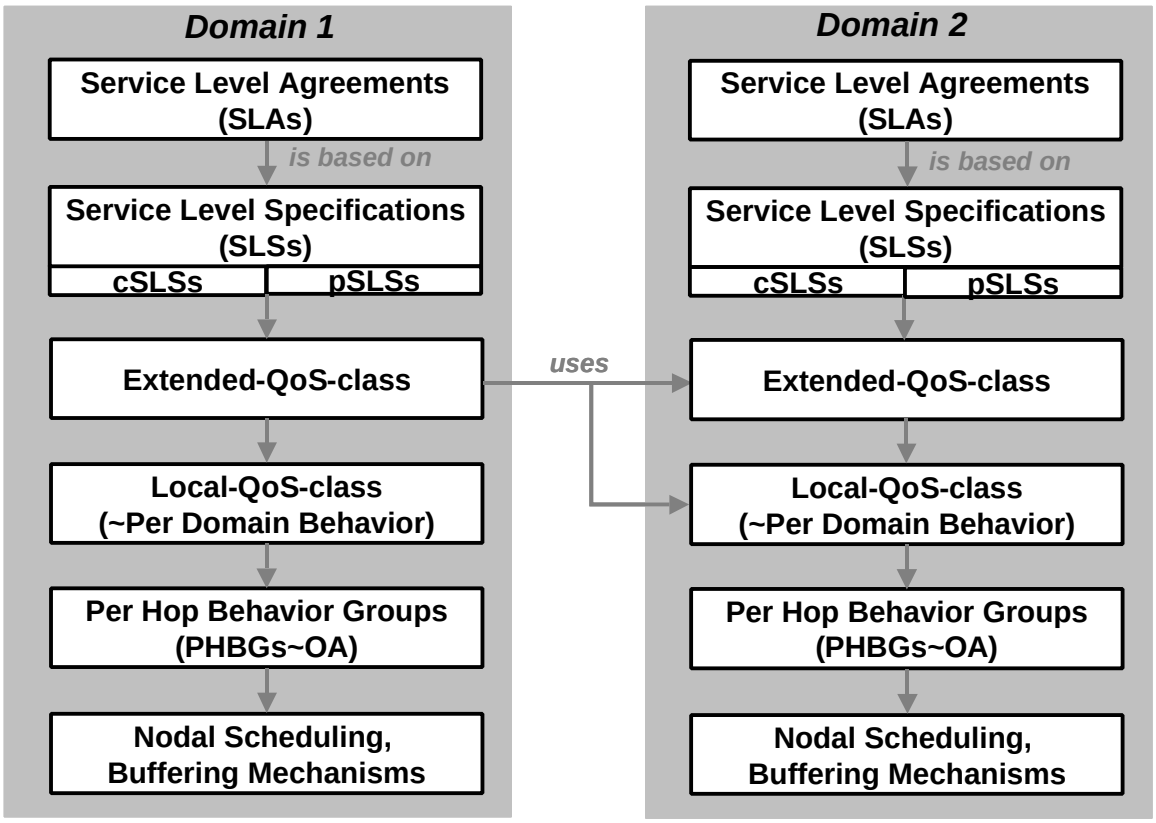


Figure 5: The MESCAL Internet QoS service model.

The MESCAL QoS service model is a layered peer model.

The essence of the model is the notion of QoS-class introduced in the previous section. Considering a provider domain, QoS-classes abstract the elementary nodal QoS enabling capabilities into sets of network-wide packet transfer capabilities, which are deemed appropriate to support the connectivity requirements of QoS-based services and applications. The notion of the QoS-class provides the necessary abstraction level for (a) building QoS-based services and (b) for linking service-peering provider domains to the end of expanding the geographical scope of their QoS-based services, independently of the underlying network-level capabilities, even technologies, employed in the different provider domains.

In particular, the layered aspect of the model refers to within a provider domain; through a 'is-based-on' relationship builds from the elementary nodal QoS enabling capabilities (IP DiffServ is assumed) to SLAs. The peering aspect of the model refers to between two provider domains; through a 'uses' relationship between QoS-classes (in the sense of section 4.2.2.4) allows different providers to combine their QoS transfer capabilities to the benefit of extending their QoS-based services beyond their geographical span.

4.4 Operations for Building Internet QoS-based Services

Following the concepts and notions of the proposed QoS-based service model, this section outlines suitable operations, called *QC-operations*, which need to be performed by provider domains to the end of building QoS-classes, and therefore corresponding QoS-based services, spanning beyond the reach of their domain (cf. extended-QoS-classes, section 4.2.2.1). It should be stressed that the purpose of QC-operations is to build extended-QoS-classes, not to actually provision –fulfil, assure- extended-QoS-classes. The identification of such operations is useful for a number of reasons:

- It puts the proposed concepts and notions into a sort of 'functional order', thus contributing to the validation of the model from functional perspectives.
- It contributes to the drawing of a functional architecture, per and across provider domains, for QoS-based service provisioning/delivery in the Internet. The identified operations should be reflected in appropriate functional blocks and/or protocol features.
- It introduces appropriate terminology against which different solutions for QoS-based service delivery in the Internet could be described and compared. Such different solutions may employ different means –functions, algorithms and protocols- in realising the identified QC-operations.

The following point is worth noting. QC-operations prompt for distinguishing and functionally decoupling the required functionality (traffic engineering and service management functionality) per provider domain for QoS-based service delivery, into intra- and inter-domain. QC-operations primarily imply inter-domain functionality as they target at building external-QoS-classes. Intra-domain type of functionality is mainly implied by the delivery of local-QoS-classes, which are taken for granted from the perspectives of QC-operations.

Considering a provider domain wishing to provide e-QCs onwards, from its domain to destinations outside its domain, the identified QC-operations are depicted in Figure 6 and described in the following sections.

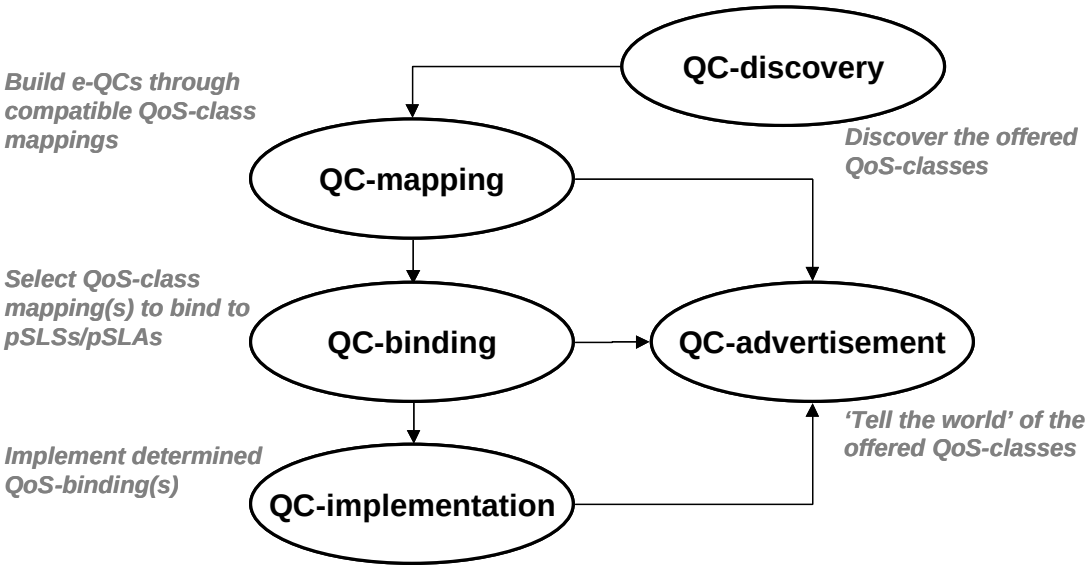


Figure 6: MESCAL QoS-class operations.

4.4.1 QC-advertisement

Through the QC-advertisement operation a provider domain informs other providers of its QoS-class capabilities. QoS-classes may be advertised at various levels as deemed appropriate by relevant policies of the provider. They may be advertised when are first conceived as a result of the marketing and service planning activities of the provider or during when the necessary actions for building them (agreements, configurations) are being taken or after they can be actually supported and provided.

As outlined in section 4.2.2.4, QoS-classes can be made known to other providers through (one or both of) the following two methods: by advertising corresponding QoS-based services (cf. pSLSs, section 4.2.1) and/or by appropriate capability announcement means. Without loss of generality, it is assumed that the advertised QoS-classes are of offered-QoS-class type (cf. section 4.2.2.3).

The means for other providers to actually use the QoS-classes advertised by a given provider as well as any other information deemed appropriate to accompany QoS-class information (e.g. topological scope constraints, corresponding Meta-QoS-Class) are assumed that they are conveyed as part of QC-advertisement method employed. Given a provider domain, the means for actually using the advertised QoS-classes should be such that they can be feasibly realised at a packet level through standard capabilities of the IP layer (see discussion in section 4.4.5).

4.4.2 QC-discovery

Through the QC-discovery operation a provider domain is able to locate and find out the QoS-classes offered by other provider domains. The discovery means should be in accordance to the means employed by providers to advertise the QoS-classes they offer.

4.4.3 QC-mapping

Through the QC-mapping operation a provider domain sees how to build extended-QoS-classes, that is QoS transfer capabilities with reach beyond its domain. This is done by determining suitable - according to the performance characteristics of the extended-QoS-class to be built- combinations of the domain's own capabilities (local-QoS-classes) with the QoS-class capabilities offered by other provider domains. The latter capabilities are made known through the QC-discovery operation (cf.section4.4.2). The combinations might be based on any grounds of compatibility deemed appropriate by the provider domain to build the extended-QoS-class e.g. based on Meta-QoS-Classes equivalence or global-QoS-class conformance criteria. To this end, the QC-mapping operation may

entail a *QC-classification* process, whereby a provider domain may classify its local-QoS-classes against widely accepted service categories e.g. Meta-QoS-Classes.

It should be noted that for an extended-QoS-class deemed necessary to be provided, a number of combinations could be potentially made. For example, this may be the case when the provider domain provides more than one local-QoS-class for the same Meta-QoS-Class. The QC-mapping operation determines a subset of the compatible combinations that could be possibly made. The term *QoS-mapping* is used to denote a 'compatible' QoS-class combination determined by the QC-mapping operation for building a particular extended-QoS-class capability.

The operation is primarily instigated by the business policies of the provider domain determining the performance characteristics of the extended-QoS-classes that need to be provided and various constraints regarding combination/service peering options.

The QC-mapping operation is denoted by the symbol '$\rightarrow$'.

4.4.4 QC-binding

As already outlined, the QC-mapping operation in a provider domain may result into a number of possible QoS-mappings for building a particular extended-QoS-class. In the general case, these mappings may involve a number of different local-QoS-classes each combined with a number of offered-QoS-classes from other -one or more- provider domains.

Through the QC-binding operation, a provider domain decides which of the possible QoS-mappings determined for building an extended-QoS-class will be used for actually providing this extended-QoS-class. The selection of using a QoS-mapping is substantiated by negotiating corresponding pSLSs/pSLAs with the provider of the offered-QoS-class pertinent to the QoS-mapping; thus 'binding' the local-QoS-class with the offered-QoS-class to the terms and conditions underlying the use of the offered-QoS-class. In other words, the QC-binding operation selects a subset of QoS-mappings to cast them into pSLSs/pSLAs with the corresponding service-peering providers. The term *QoS-binding* is used to denote a QoS-mapping for which a pSLS/pSLA with a service-peering provider has been established.

QoS-binding selection should take into account the provisioning requirements of the extended-QoS-class (e.g. in terms of maximum targeted bandwidth and cost) as well as the constraints underlying the use of the offered-QoS-classes as set by their providers (e.g. availability, cost). The latter constraints could be made available through the QC-advertisement operation. In any case, they are deemed as subjects of negotiation.

It should be noted that the QC-binding operation might result in a number of QoS-bindings for a given extended-QoS-class. QoS-bindings with the same service-peering provider may differ in the local-QoS-class and subsequently in the offered-QoS-class they use. Alternatively, QoS-bindings may differ when established with different service-peering providers. Providers may find such multiplicity advantageous for avoiding to be bound to a specific QoS-capability of a particular service-peering provider and/or exploit the merits of dynamic, multi-path routing –note that different bindings imply different intra- and inter-domain routes in general.

Related to the above, the decision as to which of the established QoS-bindings will be *put in effect* in the network for actually implementing an extended-QoS-class as well as related routing/forwarding decisions fall into the realm of (inter-domain) traffic engineering. For instance, depending on the capabilities of the IP layer and corresponding policies, a provider domain may decide to put in effect only one of the determined bindings at a time, switching to another one should appropriate conditions warrant so. Or, a provider domain may decide to put in effect all determined bindings and employ a dynamic routing scheme with or without multi-path and load distribution features.

Once in the context of an extended-QoS-class the appropriate bindings have been determined, established with service-peering provider domains and effected in the network, the extended-QoS-class capability can actually be provided. The provider domain may make known this capability to

other provider domains or customers by defining appropriate offered-QoS-classes and advertising them through the QC-advertisement operation.

The QC-binding operation is denoted by the symbol '⊕'.

4.4.5 QC-implementation

Through the QC-implementation operation, a provider domain implements at the network layer a QoS-binding. The operation encompasses only the necessary configurations at the IP layer required for the appropriate treatment of the packets. As stated in the previous section, routing and forwarding issues are outside the scope of the QC-operations.

Considering a provider domain offering a given QoS-class, which corresponds to an extended-QoS-class of the domain actually implemented through a particular QoS-binding, which in turn, by definition, involves a local-QoS-class of the domain and an offered-QoS-class of a service-peering domain, the QC-implementation operation encompasses the following aspects:

- Identification of the QoS-class according to which the packets entering the provider domain should be treated.
- Enforcement of the corresponding local-QoS-class in the provider domain.
- Enforcement of the use of the corresponding offered-QoS-class in the service-peering provider domain, which at the end corresponds to a local-QoS-class in that domain.

The above aspects should be realised based on the capabilities of the network layer –IP DiffServ/MPLS-capable routers are assumed.

Packets entering a provider domain are identified as belonging to an offered-QoS-class of the domain based on their IP header information. Similarly, the means for a provider domain to enforce the use of a QoS-binding-related QoS-class in a service-peering provider domain should be based on information contained in the IP header. For scalability reasons, the IP header information used for these purposes, should not be too fined-grained e.g. specific to customer contracts (cSLSs). Information facilitating traffic aggregation should be used e.g. DSCP.

Enforcement of local-QoS-classes is realised by mapping them (that is, the classified packets) to an OA (Ordered Aggregate). An OA corresponds to the notion of PHBG, the QoS building block of IP DiffServ domains, which prescribe to particular types of nodal packet treatment (EF, AF1-4, BE). At a packet level, an OA corresponds to specific information in the IP header, the so-called DSCP and is realised through appropriately configured scheduling (and buffering) mechanisms available at the network nodes. Note that different provider domains may use different DSCPs for the same OA.

The choice of OA per local-QoS-class should be made in accordance to the targets (bounds) of its performance parameters. When a provider domain institutes its local-QoS-classes, a set of possible OAs is associated with them for their implementation; the first OA denotes the most appropriate OA and the other OAs denote alternatives of superior performance e.g. EF could be set as an alternative of an AF1 OA which is deemed the most appropriate to implement a local-QoS-class. The QC-implementation operation determines which of the associated OAs is 'best' to be used for enforcing the local-QoS-classes, according to actual network status and state conditions.

The above aspects may be realised through the classification and marking mechanisms prescribed by the IP DiffServ architecture, or by setting-up LSPs across domains or a combination of them. The actual realisation means are left to the individual solutions for Internet QoS-based service delivery.

5 INTER-DOMAIN QOS ISSUES

5.1 Introduction

This chapter discusses a number of issues that arise in Inter-domain QoS delivery. The topics addressed cover all aspects of the MESCAL project, including peering arrangements, service guarantees, traffic engineering, scalability and multicast. The objective is to provide background information on and to explore the intrinsic aspects of each topic. Later chapters discuss these issues in the context of a solution for delivering Inter-domain QoS.

5.2 Inter-domain Peering

5.2.1 Cascaded vs. Centralised Approach

Within the MESCAL project, two major approaches have been considered to establish a consistent set of inter-domain peering agreements in order to construct end-to-end QoS-based services across Internet at large scale:

- The cascaded approach where a provider only negotiates pSLSs with its immediate neighbouring provider/s to construct an end-to-end QoS service. With this approach, service peers are also BGP peers.
- The centralised approach where a provider negotiates directly with an appropriate number of downstream providers to construct the service. With this approach, service peers may not be BGP peers.

The following two sections provide a description of these two approaches. It should be noted that the type of inter-domain peering impacts the service negotiation procedures, the required signalling protocols, the QoS binding, and path selection.

5.2.1.1 The Cascaded Approach

In the cascaded approach, the QoS peering agreements are between BGP peers, but not between providers more than "one hop away". This type of peering agreement is used to provision the QoS connectivity from a customer/domain to reachable destinations when crossing several domains.

Figure 7 gives an overview of the operations in this approach. The domain AS5 supports an intra-domain QoS capability (l-QC1). AS4 supports an intra-domain QoS capability (l-QC2) and is a BGP peer of AS5. AS4 and AS5 negotiate a contract (pSLS3) that enables customers of AS4 to reach destinations in AS5 with a QoS (e-QC1). This process can be repeated recursively to enable AS3 to also reach destinations in AS4 and AS5, but at no point do AS3 and AS5 negotiate directly.

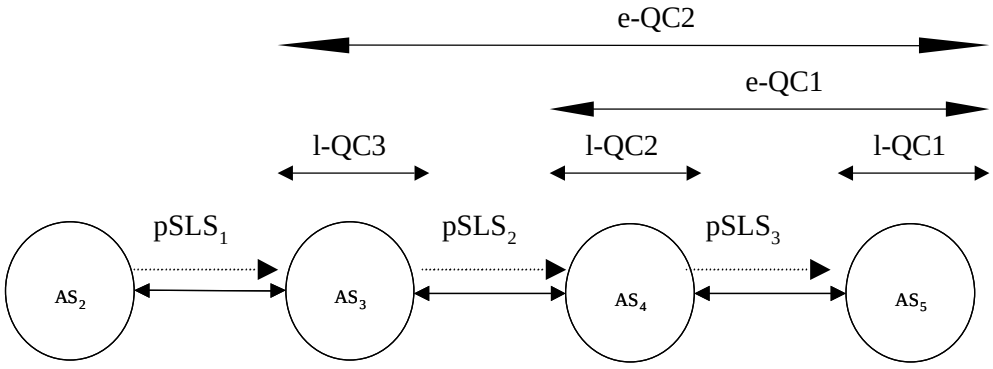


Figure 7: Cascaded Approach.

5.2.1.2 The Centralised Approach

The centralised approach disassociates pSLS negotiations from the existing BGP peering arrangements. The originating domain knows the end-to-end topology of the Internet and establishes pSLSs with a set of potential domains (neighbours, transit, and distant ASs) in order to reach a set of destinations, to offer end-to-end QoS-based services.

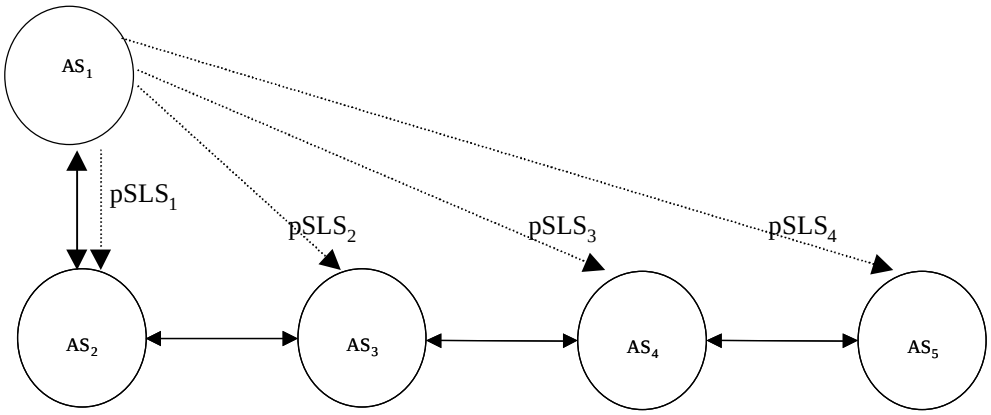


Figure 8: Centralised Approach.

The centralised approach presents an alternative to the cascaded approach providing a high degree of flexibility at the service negotiation level, but it may create deployment/scalability concerns.

Within the context of MESCAL project, we focus only on the Cascaded approach and the MESCAL solution presented in Chapter 7 is based on this approach.

5.2.2 Passive and On-demand Peering

5.2.2.1 Passive pSLS negotiation

The cascaded approach can be characterised as follows:

- The pSLS is only negotiated between two adjacent ASs, i.e. autonomous systems whose ASBR routers have established eBGP peering relationships,
- Services that are constructed by cascaded pSLSs are dependent on what has been negotiated in the downstream cascaded AS chain.

One of the concerns with the cascaded approach is that it is passive, insofar as an AS cannot directly control the QoS negotiation beyond its adjacent AS. This can be a problem if one of the ASs in the cascaded chain is not motivated to create a pSLS with a peering domain – perhaps it is not aware of the business opportunity – so the end-to-end cascaded chain cannot be created.

In order to address this problem, the pSLS On Demand is proposed and explained in the following section.

5.2.2.2 pSLS On Demand

The idea of pSLS On Demand is that an AS can request a target AS to establish a particular pSLS with one of its adjacent ASs. This mechanism assumes that the target AS can offer the desired QoS capability – perhaps it has already advertised the QoS offering – but it has not negotiated pSLSs based on its capabilities.

Figure 9 shows a scenario where it is not possible to build an end-to-end QoS agreement due to the lack of an appropriate pSLS between the AS_3 and AS_4.

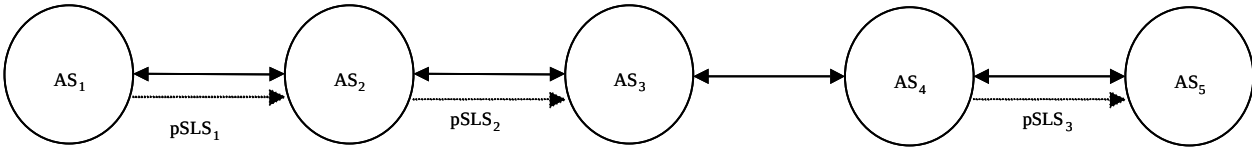


Figure 9: Passive pSLSs.

If AS_2 and/or AS_1 identify a business opportunity to build an end-to-end agreement to destinations in AS5, they can solicit AS_3 to establish the appropriate pSLS with AS4. The contents of the $pSLS_{21}$ contract include a pointer to the resulting contract $pSLS_o$.

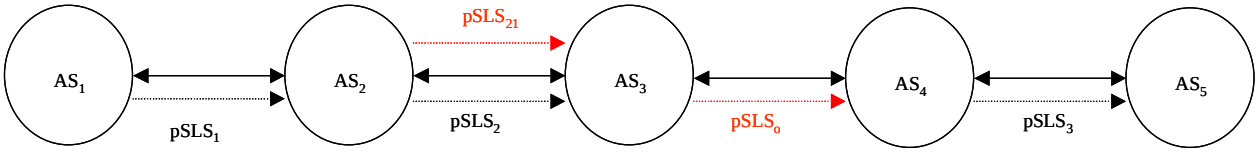


Figure 10: pSLS On Demand.

5.3 Inter-domain Service Guarantees

5.3.1 Inter-domain Service Options

It is possible to consider several service options that might be offered by inter-domain QoS-enabled IP networks, each with their particular requirements for QoS performance guarantees. For example:

- A service option that targets customers requiring differentiated network services whatever the destination of their traffic is within the domain. This service offering is only provided if the amount of the QoS traffic remains within certain limits compared to the rest of the best-effort traffic. Since the provider does not have the prior knowledge about the traffic destinations, the overall sum of flows must remain lower than the amount of resources provisioned (in the links and network elements) by the provider for this purpose. The dimensioning of the network is performed statistically and control of network resource usage may rely on monitoring. If the network dimensioning and control are performed appropriately, the end-user can expect that the QoS-enabled traffic sent in the network will reach its final destination with some loose QoS guarantees (better than best-effort). It should be noted that it would be difficult to provide strict bandwidth guarantees due to the statistical nature of the service.
- An alternative service option is dedicated to customers requiring strict QoS offering including end-to-end performance and bandwidth guarantees. To provide this service option, especially for the end-to-end bandwidth guarantee, it is mandatory to reserve the appropriate resources along the end-to-end path (over booking can also be an option) and to control the path. Traffic engineering techniques are normally used for this purpose. For example, in MPLS-enabled networks, end-to-end LSPs are established for which appropriate resources have been reserved.

The project must identify the types of inter-domain QoS Service that it will provide (see Section 7.2), as it will have a significant impact on the MESCAL solution.

5.3.2 Bandwidth Guarantees

End-to-end bandwidth guarantees can be provided to customers on an inter-domain basis but the issue requires careful consideration. For example, it is difficult to provide bandwidth guarantees to customers if the destination of the traffic is not known in advance (e.g., for services such as Internet access). However, it is possible to provide bandwidth guarantees if the destination of the traffic is known in advance, as for services such as VPNs.

5.4 Inter-domain Traffic Engineering

Traffic engineering is the means to optimise the use of available resources. Such optimisation inevitably involves the control of *outgoing, incoming* and *within the network* traffic flow. The first two are collectively regarded as inter-domain traffic engineering, while the latter as intra-domain.

The following operations are considered as Traffic Engineering (TE) issues:

- Define, provision and control local QCs (*l-QCs)*
- Reduce high variance in link bandwidth utilisation per QC
- Control of the outgoing/incoming traffic
  - Balance the traffic among external links
  - Prefer some links over the others

5.4.1 Peer Provider Selection problem

One problem that a provider is facing is the choice of adjacent ASs with which pSLSs will be negotiated. We name this problem as Peer Provider Selection.

The criteria for the selection are:

- The advertised QCs from the various ASs
- Economic criteria
  - Cost of link,
  - Cost of traffic, i.e. bandwidth per QC
- Business-oriented constraints
- The advertised network reachability information

This is in fact a cost optimisation problem. Note that the result of this selection will be more than one pSLSs for the same o-QC technical (traffic engineering) and economical reasons. These will be utilised later for load balancing.

5.4.2 Controlling the Outgoing Traffic

Load sharing is an important part of traffic engineering because it allows the traffic to spread among different paths and different classes and thus achieve better resource utilisation. We can achieve the maximum utilisation of the inter-domain resources by controlling the outgoing inter-domain traffic.

Note that the problem of optimising the utilisation of network resources requires controlling both the inter- and intra-domain resources. In this section the focus is on the inter-domain issues, but we will also elaborate on intra-domain resource control wherever is appropriate.

5.4.2.1 Load sharing based on different destination prefixes

Being able to control the load of the egress links and pSLSs on the granularity of different routes towards different destination address prefixes is an issue to be solved by traffic engineering techniques. This is to control the outgoing traffic on a per prefix basis so as to optimise the use of the egress resources.

In this simplest form of load balancing scenario we assume a single egress point, a single egress link, and a single egress pSLS that will be used to route traffic towards a specific destination prefix. Then for offering a particular o-QC we must balance the load between e-QCs (i.e., QC-binding to be put in effect)), which satisfy the requirements of the o-QC, i.e. choose an egress link/pSLS, for each destination prefix based on the different pSLSs. That means we do not require multiple simultaneous paths to be in effect for the same destination.

This exit point selection can be formulated and solved as an optimisation problem, where the objective is to optimise the utilisation of each egress pSLSs.

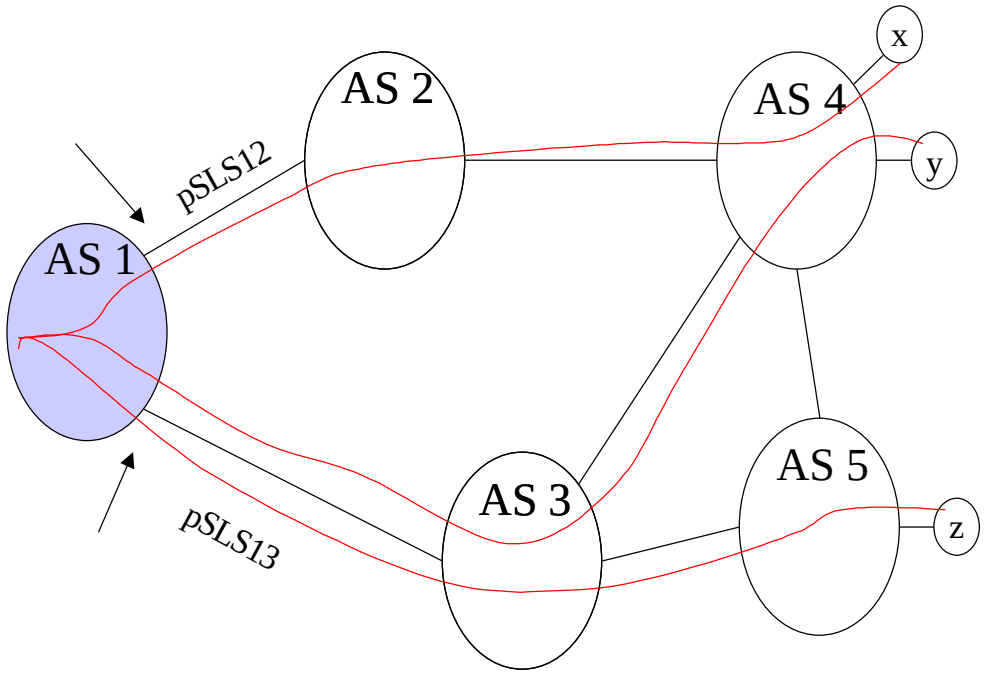


Figure 11: Balancing based on different destination prefixes.

In the simple example shown in Figure 11, we assume that pSLS12 and pSLS13 are compliant with the same o-QC and that pSLS12 has half the bandwidth of pSLS13, thus from the AS1 point of view it needs to route twice as much traffic through AS1-AS3 link as the AS1-AS2 link. In this example, the simplest way to achieve this splitting ratio is to route traffic for the prefixes towards the AS3 twice as much as the AS2. The enforcement of such load sharing decision can be done with the enforcement of specific routing policies either by fixing the path or introducing policy rules to dynamic routing protocols.

5.4.2.2 Multi-path load balancing for the same destination prefix

Load balancing on a per destination prefix basis only is not very flexible and thus the resulting engineering solution may overload one or another egress pSLSs while others are under-utilised. This situation can be improved by allowing more than one path towards the same destination prefix. In the following we will describe all possible load-balancing scenarios that can be taken when we have multiple paths towards a destination. The discussion will include intra-domain load balancing actions which are not tailored only to multi-path load balancing described in this section, but some of them may also be used in conjunction with the single path load sharing case discussed in the previous subsection.

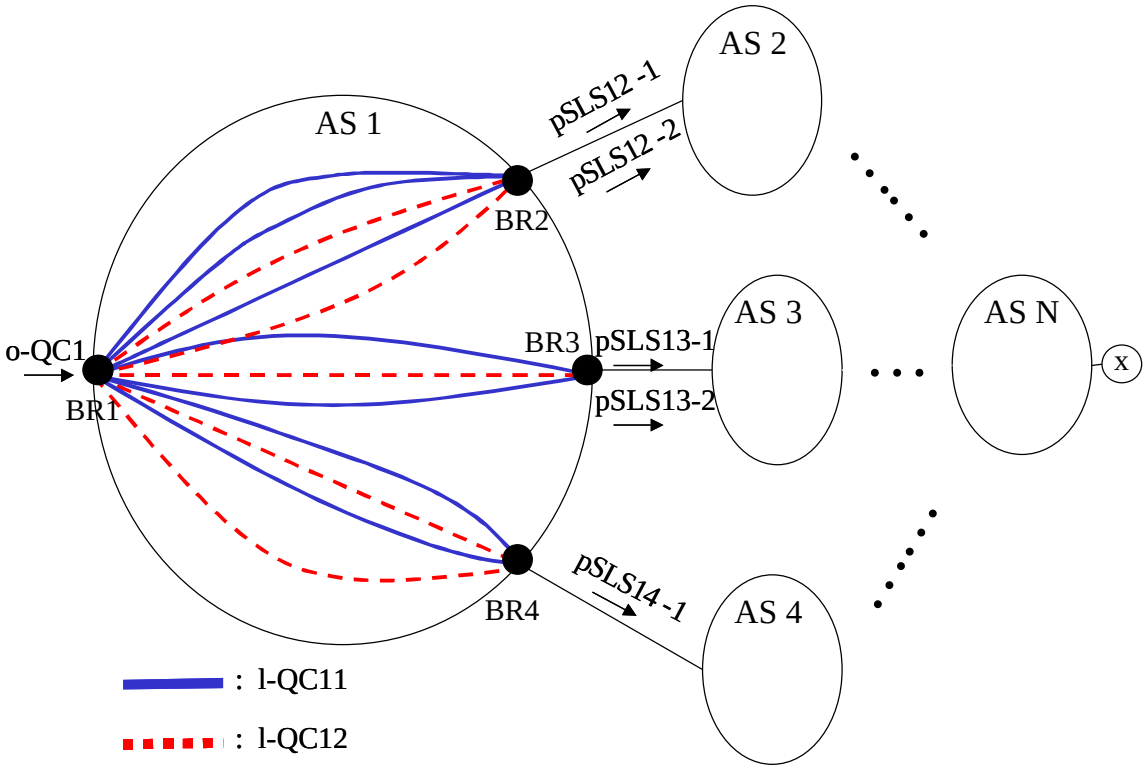


Figure 12: Load balancing possibilities (example 1).

In MESCAL, we can identify multiple levels of load balancing. In order to offer an o-QC1, we may have multiple combinations (QC bindings), which achieve the required performance characteristics of that o-QC. Among all these possibilities we have to decide:

1. Statically (*offline*) which QC-bindings and routes, i.e. e-QCs, are going to in effect for offering that o-QC (a reminder here that the o-QC is offered to an upstream AS *via* the agreement in a pSLS) so as to optimise the utilisation of resources.
2. If the previous case chooses to have more than one alternative bindings and routes in effect, then dynamically (*online*), based on measurements, we can decide for each flow which of the alternatives to use, so as to optimise the use of resources.

Static load sharing is required so that we can configure the control mechanisms in order to enforce the traffic engineering decision. The timescale is that of the Resource Provisioning Cycle (RPC) of the AS. Dynamic load sharing is required when we want to more accurately reflect on the traffic fluctuations. Note that dynamic TE is not at the per-packet timescale but rather on a per-flow or multiple flows in order of minutes.

With refer to Figure 12 and assuming that all the alternatives shown there are compatible, i.e. *as good as*, for offering o-QC1, load balancing (both *offline* and *online*) can be applied at multiple levels:

- Choosing the egress point, e.g. choosing one of the BR2, BR3, or BR4, and the egress link (see Figure 12).
- Choosing between the (potentially) multiple pSLSs, e.g. between pSLS12-1, pSLS12-2 (of course all the o-QCs of the adjacent ASs included in the pSLSs bound with the l-QCs must be “at least as good” as the offered o-QC).
- Choosing between the Local QCs (*l-QCs)*, e.g. between *l-QC11* and *l-QC12*
- Choosing between the potentially multiple paths of the chosen *l-QC*.

Note that some of the above options may not be available. As we move from static to dynamic load balancing the options may be reduced.

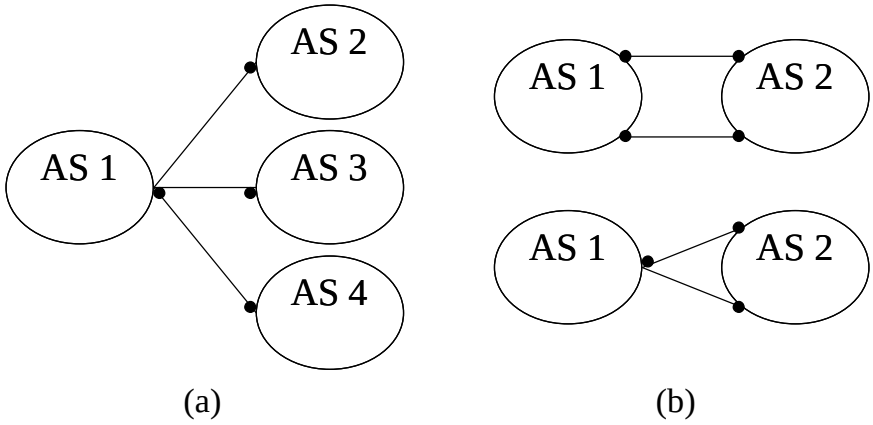


Figure 13: Choosing egress point or next-hop AS different from choosing link.

The first point in inter-domain load balancing deserves a little bit further discussion. The selection of the egress point does not mean that we necessary choose the next-hop AS and by choosing the next-hop AS doesn't mean that we choose the egress link. For example, as shown in Figure 13, in (a) when we have multiple peering at the same egress router (usually the case of multi-homed domains [Rekht95]) by choosing the egress point does not necessarily mean that we choose the next hop AS, and in (b) when we choose the next hop AS does not necessarily mean we choose the output egress link. It should be noted that in the case (a) the egress node have multiple interfaces each connected to an AS via interconnection links and in the second case (b) the AS connected to the next hop AS via multiple interfaces and interconnection links.

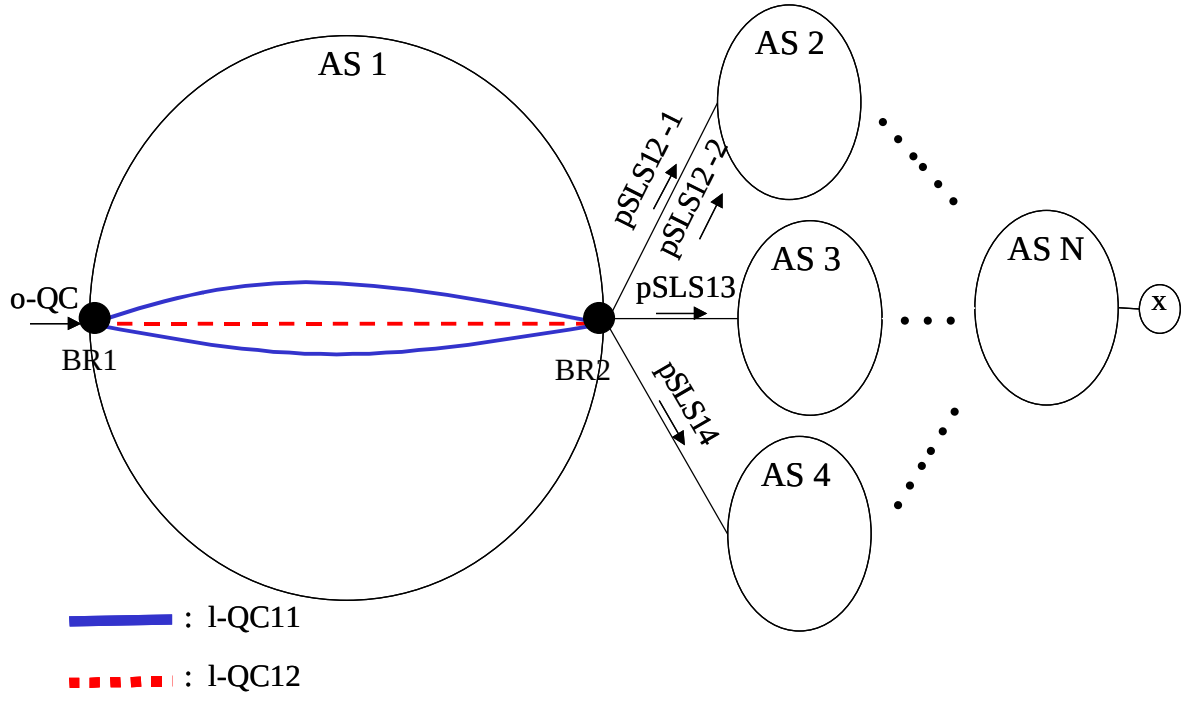


Figure 14: Load Balancing possibilities example 2.

In Figure 14, we can see that the inter-domain load balancing does not include the choice of the egress point, but includes the choice of the egress link (i.e., an output interface in the egress point/node) and the choice of the pSLS to be used. This is the scenario where an AS is multi-homed. The load balancing in this case is: a) among the interconnection links i.e., AS1-AS2, AS1-AS3, AS1-AS4, b) between the pSLSs of the chosen link, e.g. pSLS12-1, pSLS12-2, c) between the multiple l-QCs that can be bound with external o-QCs so to define multiple e-QC to comply with the offering the same o-

QC, and d) between the (potentially) multiple internal paths of the chosen l-QC. Combinations of situations as in both example 1 and example 2 may also exist.

5.4.3 Routing Aspects

The means to implement the various TE decisions is to control the routing. Even if we do not allow for balancing all the traffic as described in the previous section, routing has to be controlled in order to adhere to the QCs, both internally (l-QCs) and externally (pSLSs, i.e. o-QCs).

In this section, we will describe the requirements and the possible implementation mechanisms for controlling the inter-domain aspects of routing. Controlling intra-domain routing in order to achieve certain objectives is a very important issue, which has been studied extensively in the past [TEQUI] [Fortz00] and is not the main focus of MESCAL.

In the cascaded approach, inter-domain routing has the following aspects:

- Choose the egress point from the AS
- Choose the next hop AS
    - Choose among the possible links

5.4.3.1 Requirements for the inter-domain route selection process

There are three requirements from inter-domain traffic engineering:

1. It must be QC-aware
2. It must be constrained by the pSLS agreements
3. It should support load balancing capabilities for different destination prefixes

The first requirement says that the routing decision should be aware of the fact that the traffic that will be routed based on a particular QC. Thus the decision for routing may be different, and effectively is to have a different routing policy for each of the supported QCs. Being able to differentiate the traffic flow between different QCs is very important for the performance of the end-to-end QC. This requirement includes another important aspect that we need to "inject" somehow the o-QC information into the routing information distribution process. For example, if BGP is the protocol used to distribute the routing information, then we need to have the appropriate attributes for disseminating the QC information, which will be processed by the BGP peers.

The second requirement states that the possible egress points for specific QCs are only the ones for which we have agreed some pSLS with a downstream peering AS. This means that even if we have classical NLRI information (i.e. for best effort traffic) through some peering AS, we cannot use that AS as the next domain for QoS traffic, if we do not have a pSLS for a using particular external o-QC. A consequence of this requirement is that each time we agree on a new pSLS with a downstream peer we need to make this information available to the route selection process.

The third requirement reflects the discussion of section 5.4 on balancing the load over the multiple egress points in order to avoid overloading some of them, while others are under-loaded. This balancing is performed over *different destination prefixes*.

As a secondary requirement for routing:

4. It should support load balancing over multiple egress paths (as described in the previous section) for the *same destination prefix.*

Although the fourth requirement is important when we want to perform traffic engineering, we leave it as a "should", indicating that it is important but not a mandatory feature. Ideally we would like to have the flexibility to perform load balancing over non-equal cost paths with non-equal sharing ratios, but, if this is not easy to support from the implementation point of view, then we can make use of equal-cost traffic splitting. The exact load balancing capabilities are of great importance when we are to devise the Traffic Engineering algorithms. It is envisaged that there will be a trade-off between the

additional required protocol changes and the flexibility and optimality that can be achieved by the traffic engineering processes.

The above can be realised through static or dynamic routing schemes (cf. Discussion in Section 5.4.3.3).

5.4.3.2 Propagation of Inter-domain QC routing information

The proposal in MESCAL is to keep BGP as the basic means for propagating Network Level Reachability Information (NLRI) on per QC basis. The basis for this work will be the IETF initiative for defining the QOS_NLRI [Crist02] together with the work which allows the advertisement of multiple routes towards the same destination prefix [Walton02].

BGP provides the means to influence to whom and from which ingress points the routing updates are to be sent/received by filtering the updates according to some policies. These policies will be enforced so that to advertise the QoS reachability only to peers with which we have agreed on with some pSLS.

The appropriate attributes, which describe the QCs must be defined and included in the advertisements. This may be a source of some scalability concerns since it will lead to increase the routing table size depending on the number of the supported QoS classes, which in the MESCAL solution is bound by the number of DSCPs i.e. 64.

The QC information is propagated by BGP whenever a pSLS is agreed. One can allow for the QC information to change more dynamically, e.g. at each Resource Provisioning Cycle (RPC), in order to achieve some kind of QC performance monitoring. Although this feature may be quite useful for the engineering (e.g. load balancing) of upstream ASs, it may constitute sources of instabilities. This option has to be examined in greater detail in the context of the project to assess the potential instabilities.

The implementation overhead is related to the definition and manipulation of the attribute to carry the QC information. Note that modifying a BGP route selection process may be risky as the BGP Finite State Machine (FSM) may be affected accordingly.

5.4.3.3 Enforcing the inter-domain routing control policies

The basic requirements of inter-domain traffic engineering can be met with either fixed or dynamic path routing solutions. As stated previously, in all cases BGP is used, for the dissemination of QC-aware NLRI information.

The route selection algorithm should take into account the QC information and it should perform load balancing on the exit links (and pSLSs) for traffic destined to different prefixes. The latter is the first case for load balancing as described in section 5.4.2. Load balancing over multiple paths to the same destination prefix is an extra non-mandatory feature.

5.4.3.3.1 Fixed path routing

Enforcing inter-domain traffic engineering policy for statically fixing the path can be implemented in two methods:

1. If we assume the IGP-EGP model enforcing the fixed path routing, it means just to add the routing information into the BGP, whenever a pSLS is agreed (including renegotiations). We have to inject the new route and enforce the appropriate policies so that to advertise only the selected routes. In addition the BGP route selection algorithm has to be overwritten to ignore route changes advertisements (i.e. fixing the path).
2. The second method is to implement the fixed path routing decisions by using the mechanism proposed by CISCO in the Internet Draft, “Inter-AS MPLS Traffic Engineering” [Vasse03]. Here, we describe the use of “Scenario 1: Per AS Traffic Engineering Path Computation” solution as described in the draft. The use of “scenario 2: Path Computation Server” solution is for further study.

In the solution of Scenario 1, there must be the support of inter-AS TE paths, spanning more than one domain. In some cases, this solution can be used to support the establishment of end-to-end TE paths. The MESCAL solutions described in Chapter 7 are based on the cascaded approach, which require service peering relationships *only* between adjacent domains. Thus, we will not take advantage of the full spectrum of the Cisco's solution capabilities, but rather we use the mechanism specified in the Scenario 1 where we set-up TE paths towards and up to the first adjacent AS. This is shown in Figure 15, where the TE paths are set-up to the first ASBR of the adjacent AS. Note that although the Label Switched Paths (LSP) are set-up to a ASBR2 and ASBR4, this does not impose any administrative problems since the label switching operation can stop at egress ASBR, i.e. ASBR1 and ASBR3, due to the penultimate hop label popping feature of MPLS.

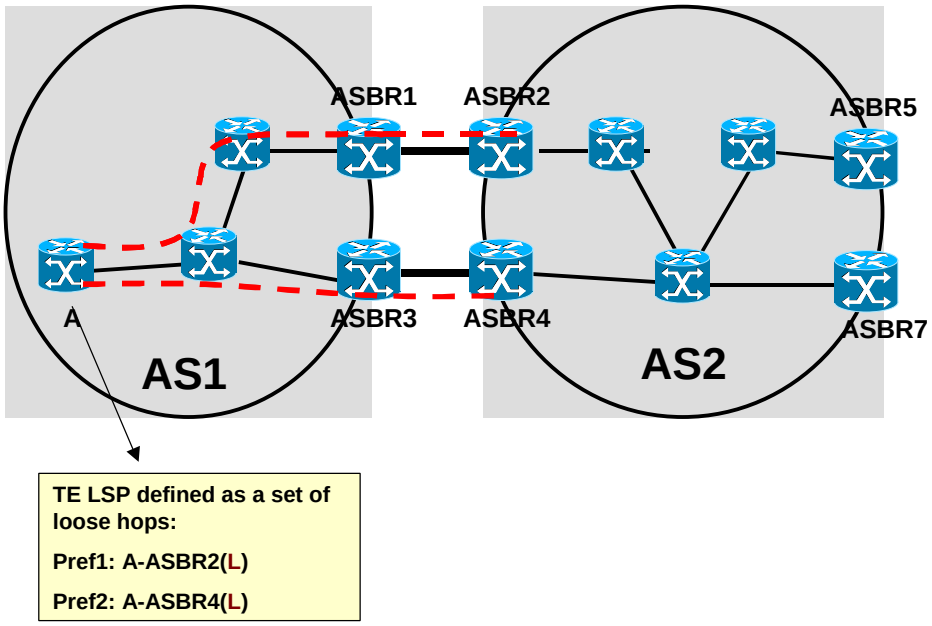


Figure 15: Facilitating the CISCO inter-AS solution scenario 1 proposal.

The CISCO solution proposes to flood the TE information related to the ASBR-ASBR link(s) even though there is no IGP enabled over those links. This allows the TE DB (Data Base) in each router to include TE information (TE metric, bandwidth, etc.) for the ASBR-ASBR links and thus to the potential head-end Label Switched Routers (LSRs). Since it is required for the inter-domain traffic engineering to be QC-aware, this means that the TE information must be on a per class of service, e.g. per {TA}PSC according to the definition provided by [LeFau03a] where TA is Traffic Aggregate and PSC is PHB Scheduling Class..

There are three important considerations in order to use this mechanism for enforcing the inter-domain TE decisions. We need to be able to manipulate the per-QC TE information (metric) of the ASBR-ASBR links flooded by the link state IGP. The second consideration is that we may need to change or overwrite the result of the CSPF (Constraint-based Shortest Path First) algorithm for computing the TE route. The final consideration has to do with implementation of the load balancing over multiple paths. This is supported since the solution allows for the definition of multiple paths, even for the same destination address prefix, and the ability to map traffic onto the multiple paths [Zhang03].

5.4.3.3.2 Dynamic path routing

Static path routing can support the required functionality in order to enforce the inter-domain TE and QoS policies discussed in this document. However, such routing schemes are hindered by lack of adaptability to changing topological and/or load changes and the restricted potential in achieving load-balanced states and thus optimising network utilisation, as compared to dynamic and/or multi-path

routing schemes. The following questions should better be addressed by the employed routing scheme: - how to learn QoS path failures? - and how to respond to such failures?

A dynamic inter-domain routing protocol, i.e. BGP, extended to convey QC-related information (we name this protocol as qBGP), can be used to answer the above questions, since link failure detection is an implicit capability of IP routing protocols like BGP. QC-related information can be conveyed by the BGP UPDATE messages based on the results of the engineering processes of a RPC. It is expected that the supported QCs in one AS will not differ considerably from one RPC to the next, and thus the information injected into BGP will not change considerably.

In general, the changes to BGP will be similar to the first case (see section 5.4.3.3.1) for implementing the fixed path routing approach. The BGP path selection algorithm needs to change in order to take into account the QC information. The details of the qBGP extensions mentioned above and the potential instabilities of the dynamic routing behaviour will be studied in the course of the project, and the starting point will be the initiative for the QOS_NLRI [Crist02] attribute, which is able to convey QoS information between domains and the work which allows the advertisement of multiple routes towards the same destination prefix [Walton02]..

5.5 QoS Issues

5.5.1 The "QC splitting" Problem

Figure 16 shows an example where pSLSs have been established between adjacent domains allowing each domain to send QoS-enabled traffic to its peering partner for crossing the Internet. Different l-QCs have been defined and deployed within each domain.

Users C1 and C2 requested red and blue e-QCs where each ordered set of l-QCs (red and blue) represents an e-QC i.e., red e-QC: (QC12, QC21, QC54, QC74) and blue e-QC: (QC11, QC21, QC51, QC71). If the DSCP field in the IP packet header is used for QoS-signalling across domains, both red and blue e-QCs are mapped to QC21 at AS2. At the AS2 egress point/s, we will encounter a splitting problem in that it will not be possible to distinguish the red and blue packets, based on DSCP values, so as to re-mark them with their individual DSCP values for onward transmission.

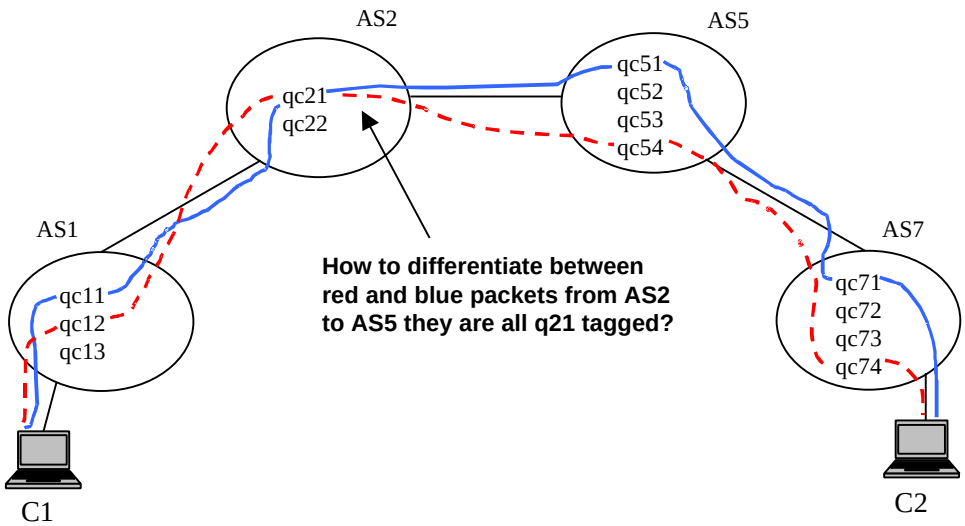


Figure 16: The QC splitting.

The QC splitting problem arises when a provider binds more than one o-QC of a service peering domain to one of its l-QCs. The issue is that:

"What should be and how to determine the appropriate DSCP marking for the datagrams forwarded to AS5?"

Any inter-domain QoS solution must overcome the QC splitting problem, while controlling the amount of state information that must be stored at each ASBR. To solve the splitting problem, the egress point at the AS could use one of the following mechanisms:

- By using full-set or sub-set of the 5-tuple (source and destination IP addresses, protocol, source and destination ports) and the DSCP to be used in the AS. Building such a list/table would assume that all cascaded ASs should know about c/pSLSs (at the aggregated level) and/or the QoS classes supported by the neighbouring domains.
- By employing a source route descriptor, embedded in the IP packet (*e.g.* IP source route options), which would explicitly state the ordered set of DSCP. This descriptor would be populated by the source or by a device close to the source.
- By using virtual QCs, i.e. map the flows belonging to the red and blue e-QC to different DSCPs in order to avoid the splitting problem. The treatment of both flows within the AS must be the same i.e., the same PHB and intra-domain route will be used for both.

5.5.2 IPv6 Issues

It is an objective of the MESCAL project that its solution should be applicable to both IPv4 and IPv6 networks. This is facilitated by the common approach to DSCPs for example, as discussed below. However, there are some IPv6 features mentioned below that can be exploited to enhance further the proposed solution.

The definition of QoS (cf. DiffServ) has been integrated in the specification of the IPv6 protocol. [RFC2460] defines the 8-bits field called "Traffic Class" allowing services differentiation as defined in [RFC2474] of Ipv4. This field, commonly known as the DiffServ (DS) byte, is composed of two parts like in IPv4 (DSCP and the two ECN bits). Therefore, the 8-bit Traffic Class field in the IPv6 header is to identify and distinguish between different classes or priorities of IPv6 datagrams.

In other words, the Traffic Class field in the IPv6 header is intended to allow similar functionality to be supported in IPv6 as in IPv4 DS field bits.

IPv6 facilitates traffic engineering approaches that are not possible with IPv4. For example,

- Exploiting of the Flow Label field: The Flow Label field is a 20-bit field included in every IPv6 datagram header. Datagrams are labelled by the source to identify a flow. An intermediate router can use this value to apply a specific treatment to the datagrams. To enable flow-specific treatment, flow state needs to be established along the path from the source to the destination. Within the context of the proposed MESCAL solution, this field could find an interesting applicability. For instance, one possibility offered by the Flow Label could consist in using it as an extended DSCP field using 20-bit length.
- Defining extension header(s): In IPv6, optional network-layer information is encoded in separate headers that may be placed between the IPv6 header and the upper-layer header of a datagram. There are a small number of such extension headers, each identified by a distinct Next Header value. New headers can be defined in order to implement a new service or option, without modifying the core IPv6 protocol specification. Regarding inter-domain QoS, some information exchanges or some mechanisms could take advantage of this IPv6 feature.

5.5.3 Ingress/Egress Conditioning

Solutions for inter-domain QoS that require complex traffic conditioning at ingress/egress points need to be aware of the capabilities/limitations of the high speed ASBRs. Otherwise, QoS solutions may place more functional demands on these routers that cannot be feasibly sustained. MESCAL will take this factor into account when assessing its solution.

When crossing multiple domains, each flow must be treated based on the l-QCs selected for that flow in each domain. To the end of eliminating scalability problems (see discussion in next section), aggregate-level information contained in the IP header, notably the DSCP field, should be used for the QC-signalling. It may be necessary to re-mark the packet's DSCP at the ingress point of the AS to the appropriate DSCP value (l-QC) and re-mark to the another DSCP value at the egress point of AS. Figure 17 shows this operation, where packets arriving at the transit domain with DSCP1 are re-marked with DSCP45 for transit and re-marked to DSCP1 at the egress.

Current routers are capable of re-marking DSCPs and/or performing traffic conditioning at ingress/egress interfaces but on high-speed interfaces, the process for traffic conditioning functionality must be simplified. .

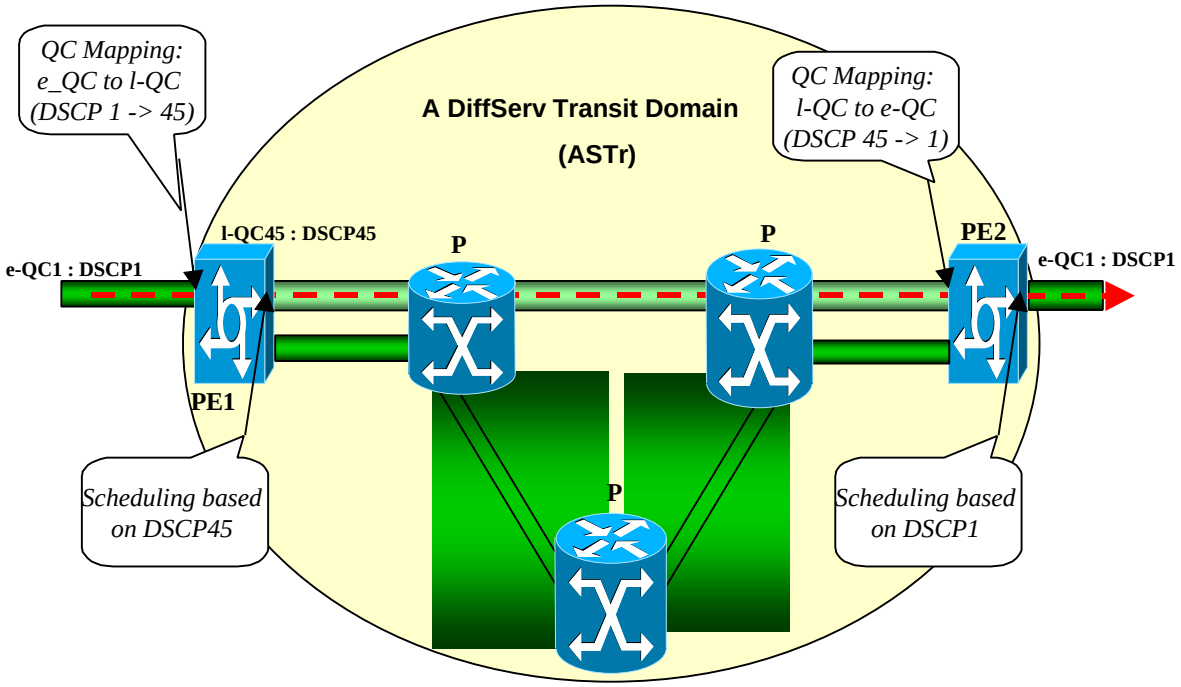


Figure 17: Ingress/Egress traffic conditioning.

5.6 Scalability & Complexity Issues

5.6.1 QC Implementation Issues

With the QC enforcement we mean the process of implementing QC-bindings (cf. Section 4.4.5) – classifying, enforcing, and forwarding of QoS-enabled packet to the correct paths (IP routes or LSP paths), as appropriate to the o-QC treatment that these packets should receive per domain. QC enforcement takes place at the data-plane after the c/pSLSs have been established. Specifically, QC-bindings are realised by downloading appropriate information for setting up the traffic classification and marking mechanisms of the DiffServ-capable routers and QoS-based packet forwarding is performed by accessing the QoS-based routing tables. QC-signalling is performed across all domains using DSCPs/MPLS-EXPs.

The following section explores different options regarding the use of IP header information in realising QC enforcement. These options justify the expected trade-off between increased flexibility in implementing QC-bindings and corresponding forwarding decisions, and per-packet processing overhead in the routers.

5.6.1.1 QC Implementation in MPLS-Based Networks

In tunnel-based solutions such as MPLS, the process of QoS-based packet routing must take place at the head-end of the tunnels. Provision must also be made at ingress boundary routers for QC enforcement when inter-AS MPLS is used. QoS-based routing is to direct specific traffic to the specific tunnel. In addition, traffic belonging to MPLS tunnels should receive different PHB treatment along the tunnel path depending on their QCs. The MPLS-EXP is the only visible field along the path to be used for service differentiation and for directing each tunnel's traffic to specific queues/schedulers. The issue is whether a unified set of EXP definitions is used across all domains or there is a need to remark the EXP at the ingress point of each AS.

5.6.1.2 QC Implementation in IP-Based Networks - Scenarios

Routing protocol should normally provide information for packet forwarding by taking into account the packet's associated l-QC. But before inter-domain QoS packet forwarding occurs, the packet's DSCP must be mapped and set to an appropriate value. Both IGP and EGP protocols for routing purposes should be QC-aware. In the following scenarios, we take into account the actions required at the AS ingress boundary routers for QC implementation - QC enforcement and packet forwarding.

In Figure 18 to Figure 21, each customer network administratively belongs to its directly connected ISP/AS (e.g., N1 to ISP/AS1). It is assumed that unique l-QCs are used in each domain. Thus, two distinct flows originated from two different sources within an AS (e.g., N0 and N1 in Figure 18) using the same e-QC and destined for a destination AS, will exactly traverse the same AS path along the route to the destination.

IP address aggregation at the network-level (i.e., network prefix/length tuple) is used in the QC enforcement scenarios described in the following sections, in order to prevent to work with individual IP addresses. Further address aggregation within an AS is possible if the network addresses belong to the AS is aggregated to a higher level of aggregation and resulted to another address tuple (i.e., prefix/length). The prefix/length tuple is available to BGP routers that can be used for QC enforcement.

5.6.1.2.1 Unified QoS Classes (l-QCs) Across All Domains (Scenario 1)

In this scenario, each l-QC has the same DSCP in every domain. While this has benefit of requiring very little ingress/egress conditioning at the domain boundaries, except at the customer ingress point, the scenario is neither realistic nor flexible. It is unlikely that network operators would agree to such constrained l-QC/DSCP mapping. Additionally, the binding flexibility is severely constrained. For example, in Figure 18, QoS binding cannot be achieved between different QCs (l-QC10 and l-QC11).

QoS-based packet forwarding need to be performed based on source address, destination address and DSCP. Source inspection for packet forwarding is required because traffic from different sources going to the same destination may transit different paths based on the e-QCs.

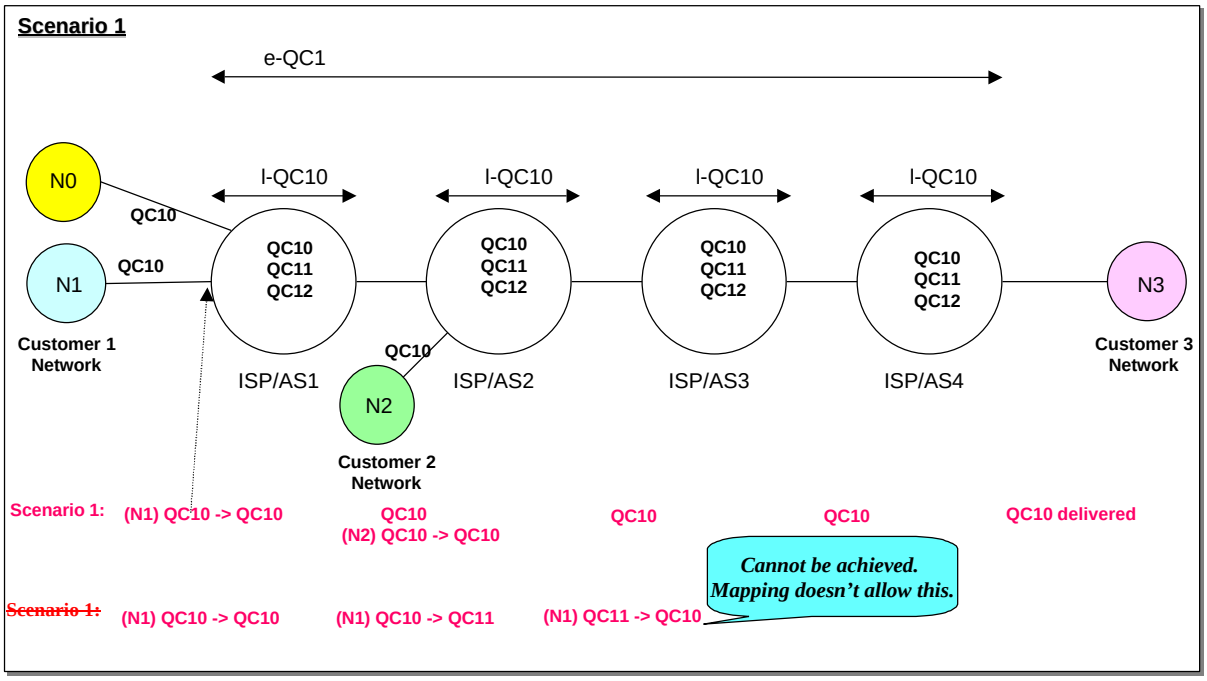


Figure 18: QC Implementation in IP-Based Networks - Scenario 1.

5.6.1.2.2 Direct l-QC Mapping (Scenario 2)

Scenario 2 differs from Scenario 1 in that DSCP manipulation is introduced at the domain ingress nodes. Each domain uses its own QoS definition using DSCP to differentiate them. Even with the introduction of this function, the binding we are still very constrained.

QoS binding is restricted as in Scenario 1 as shown in Figure 19. QC enforcement is performed on direct l-QC mapping and QoS-based packet forwarding is similar to Scenario 1.

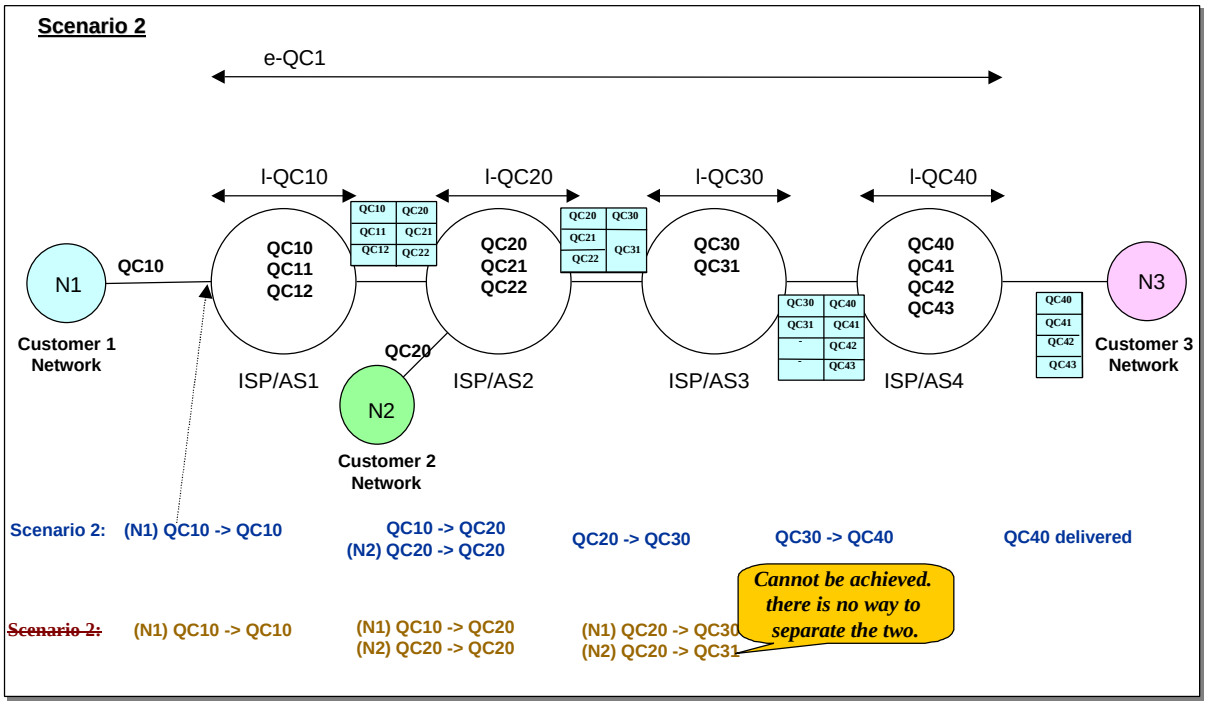


Figure 19: QC Implementation in IP-Based Networks - Scenario 2.

5.6.1.2.3 QC Implementation by Using Destination Network Address & Packet's DSCP (Scenario 3)

In Scenario 3 at the ingress point of each domain, QC implementation is performed based on the destination address and DSCP. Packet having the same destination address and DSCP are mapped to the same l-QC. However, a problem arises if, as shown in Figure 20, customers (N1, N2) will reach destination (N3) using QoS classes e-QC1 & e-QC2 respectively. The QC splitting problem, described in Section 5.5.1, arises. Introducing destination address processing in this scenario is a complexity concern, although this problem can be overcome by additional DSCP manipulation, at the domain egress point. A solution is proposed in Section 7.3.2.6 that eliminates the dependency on destination addresses as part of QC enforcement, but with limitation on the number of e-QC that a domain can support.

At ingress point of each domain, packets are examined by looking at the destination address and embedded DSCP. Packets having the same destination network addresses and DS code points are mapped to the same l-QCs. But packets coming from different sources (i.e., different ASs) possibly require different l-QCs to be mapped to, based on their e-QCs, and may use different routes to the destination. Thus, Destination Network Address & Packet's DSCP) are not adequate for proper QC implementation.

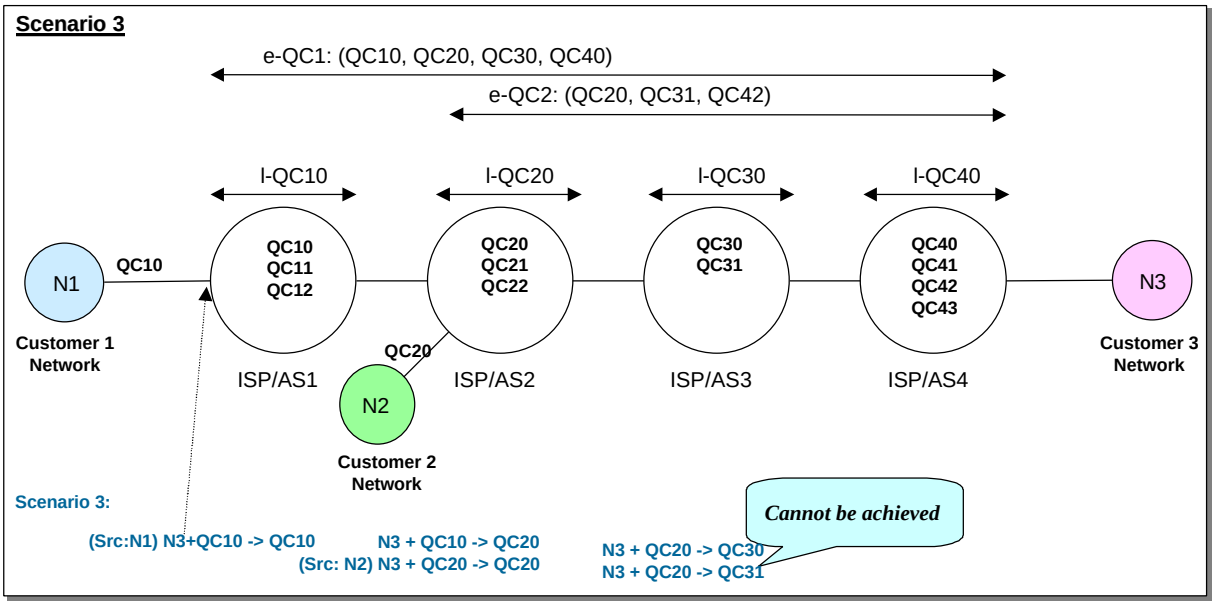


Figure 20: QC Implementation in IP-Based Networks - Scenario 3.

5.6.1.2.4 QC Implementation by Using Source & Destination Network Addresses and Packet's DSCP (Scenario 4)

The method introduced in this scenario provides total flexibility for QoS mapping and binding across all domains.

This scenario uses aggregate SLS characteristics and requires information on source network address, destination network address, l-QC used in the preceding AS and the l-QC that the packet is going to map to (see Figure 21) for QC enforcement. This requires full packet inspection (source address, destination address, DSCP) which is costly to ingress border routers.

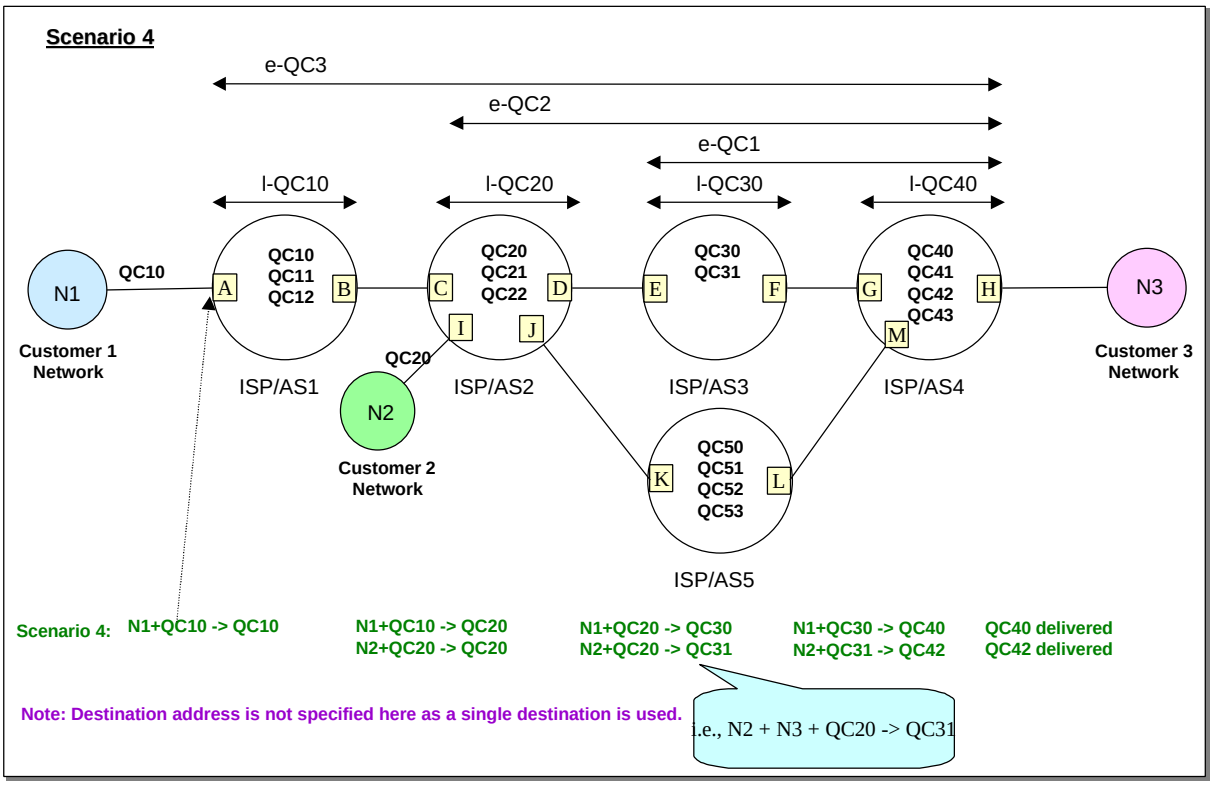


Figure 21: QC Implementation in IP-Based Networks - Scenario 4.

To perform QC enforcement on an IP packet sent from a customer in AS1-N1 to a customer in AS4-N3, the following actions are required across domains:

1- Border routers must be aware and act based on the tuple: (source AS1-N1, destination AS3-N3, DSCP embedded in the packet header, the l-QC to mapped to). In Scenario 4, the preceding AS is used instead of AS1-N1.

2- Router A in ISP1 maps the customer QoS class to QC10 by using three tuple (source IP address, destination IP address, packet's DSCP)

3- Router C maps QC10 to QC20 by using tuple (AS1-N1, AS3-N3, QC10)

4- Router E maps QC20 to QC30 by using tuple (AS1-N1, AS3-N3, QC20)

5- Router G maps QC30 to QC40 by using tuple (AS1-N1, AS3-N3, QC30)

6- Packet is directed to N3's customer via G and H routers.

Figure 22 shows an example for look up process at boarder router of an AS.

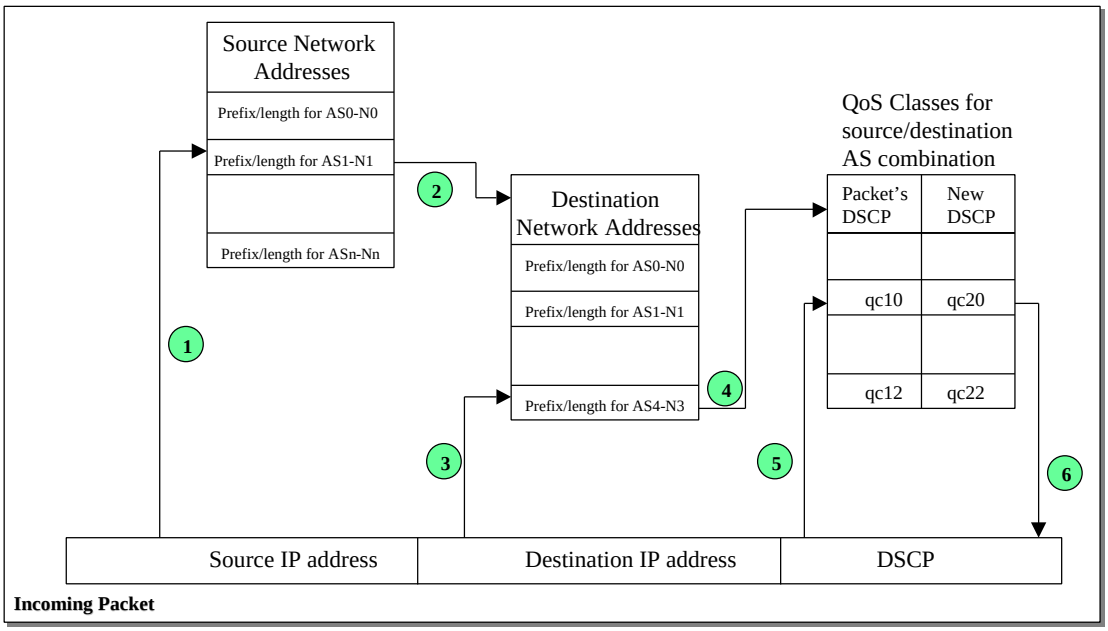


Figure 22: QoS class table lookup at router C of AS2.

BGP update messages contains a list of <prefix, length> tuples that indicate the list of destinations that can be reached via a BGP speaker. The update message also contains the path attributes, which include such information as the degree of preference for a particular route and the list of ASs that the route has traversed. In order to provide the information for QC enforcement and packet forwarding, the border router's (e.g. C) outgoing interface (similar to tunnel interface in the MPLS environment) or the outgoing boarder router (e.g., D) in the AS2 can be specified as part of this lookup. Consequently, the QC enforcement and packet forwarding can happen in this process.

5.6.2 QC Mapping & Binding

Each domain may offer a large number of l-QCs with different performance characteristics. Since there can be a large number of domains, a large number of possible/potential QoS mapping/binding can be found to satisfy the e-QCs performance targets at finer granularities. This can increase the number of possible paths to provide the e-QCs performance targets. An approach that creates many different e-QCs and possible paths may create complex routing issues and also degrade the routers' performance. This must be avoided by any proposed solutions. The number of e-QCs offered across the domains should be limited in order to avoid having complex routing tables, degrading the router performance, etc. To this end, the project has devised the notion of Meta-QoS-Classes to make this issue more manageable.

5.6.3 BGP

Any enhancement to the BGP protocol needs to be assessed for scalability and stability.

The use of BGP to carry QoS capability information between domains may lead to increasing the size or the complexity of the routing tables, as discussed in Section 5.4.3.2. The increase of the Internet routing table size has been a continual concern to routing manufacturers and the IETF. MESCAL needs to assess the consequences of its solutions on this aspect of BGP operation.

The stability of BGP is also an aspect that the project must address insofar as the frequency of updates caused by the MESCAL solution.

5.7 Multicast Implications

5.7.1 Multicast Service Models

Proper selection of multicast service models is a vital prerequisite for successful development in provisioning QoS-enabled multicast services in the Internet. It has been argued that the service model of IP multicast [Deeri88] was originally defined without an explicit objective in commercial services, which is one of the major reasons for its slow deployment [Diot00]. IP multicast, also known as Any Source Multicast (ASM), is an open group service model in that there are no mechanisms that restrict hosts from sending data to a group, or receiving data from it. In summary, the traditional IP multicast is lacking sophisticated group management. Source Specific Multicast (SSM) [Holbr03] is proposed as a closed group service model, and it has received more and more attractions ever since its birth. Compared with ASM, SSM has its own advantages in multicast source management and implementation scalability.

5.7.2 Multicast Service Level Specification (mSLS)

In IP multicast, group members are always anonymous to the multicast source. Moreover, almost all the multicast applications are receiver initiated other than sender based. Concerning QoS requirements, it is individual group members that request different service levels based on their own capabilities. These characteristics require that the Service Level Specification for QoS aware multicast applications should not be borrowed directly from the unicast scenario that is purely source based. How to define and implement multicast oriented Service Level Specification is one of the most important issues in the relevant deployment.

5.7.3 Multicast routing

Using PIM-SM [Fenner03a] with the aid of MBGP [Bates00] / MSDP [Fenne03b] has once been recognised as a promising near-term solution to the deployment of IP multicast services in the Internet. However, whether this is a valid argument is still under debate today. In this approach, PIM-SM caters for the construction of intra-domain multicast trees, and MSDP has the functionality of discovering active sources located in different domains so that these intra-domain trees can be connected together to form a unique inter-domain tree. MBGP is the multi-protocol extension to BGP4 that allows incongruent routes for unicast and multicast traffic across multiple domains. BGMP [Thale03] / MASC [Rados00] was first proposed as a long-term solution for internet-wide multicast routing, but it has not seen any significant progress in practical development till now. On the other hand, IGMPv3 [Cain02] and PIM-SMv2 have been adapted to support the SSM service model, with capabilities of source filtering and explicit group join. As far as the MESCAL project is concerned, it is also an important issue to select a proper routing infrastructure from existing schemes (MSDP/PIM, BGMP/MASC and SSM) for further QoS deployment.

It should also be noted that, since none of the existing multicast routing protocols support QoS aware routing, some adaptation/extensions would become necessary to achieve this capability. One of the typical issues is Reverse Path Forwarding (RPF) checking that is used to detect loops in multicast tree. In PIM-SM, if a multicast packet does not come from the interface, which is used to deliver unicast traffic towards the source, it will be discarded. However, the paths computed by QoS routing mechanism are often not the shortest one, and hence QoS multicast tree construction will fail if the conventional RPF checking is performed.

5.7.4 Multicast Group Management

In the IP multicast, IGMP is used to notify Designated Routers (DR) on active receivers for each group. A new group membership report will trigger the underlying routing protocol for sending the corresponding join request, while redundant reports for the same group will be suppressed. This will not be the case when individual receivers demand heterogeneous QoS requirements for the same group session. As a result, mechanisms for handling QoS-aware group membership reports will also be

investigated in the MESCAL project. On the other hand, multicast group member admission control will become a new functionality of group management, and this is another important issue for successful QoS provisioning.

5.7.5 Multicast Scalability

It has also been deemed that scalability is one of the significant obstacles that hamper fast development in multicast services. This issue exists not only in the inter-domain semantics such as AS-level source discovery and class D address allocation, but also in the group state maintenance at the level of router implementation. In the MESCAL project, when we consider the QoS enabled multicast services, efforts should be targeted at minimising extra impacts on both cases. The proposed solution should not significantly worsen the current multicast scalability problems, so that the corresponding implementation is too complicated to be achieved.

6 THE MESCAL FUNCTIONAL ARCHITECTURE

6.1 Overview

This section introduces the functionality required for the provision of inter-domain QoS services from the perspective of a single provider.

The functional architecture analyses the overall problem of providing inter-domain QoS and decomposes it into a set of finer grained components. One of the objectives of this exercise is to aid the development of our solution by breaking it down into manageable entities while maintaining a holistic view of the overall issues to be solved. In essence it is a divide and conquer exercise. Each of the functional blocks will be studied in detail in the next phase of the MESCAL project by developing suitable algorithms and protocols to implement the required functionality and to interact with the other functional blocks to provide the complete MESCAL solution. Once the functions, algorithms and protocols have been studied and specified from a theoretical point of view they will be mapped to implementable engineering blocks during the prototyping and experimental phases of the project.

Three levels of decomposition are developed in this section. The first introduces the overall aspects of the problem and solution by splitting it into five broad functional areas. Each area is dealt with in more detail in the second – intermediate – functional decomposition which is mapped to the data, control and management planes of network operation. The third level decomposes each area further into a set of co-operating functional blocks that together offer a solution to the inter-domain QoS problem within a single provider.

While the architecture describes the full set of functions required for a provider to participate in the end-to-end provision of QoS-based IP services by no means does it prescribe the implementation means by which they will be realised – within network equipment or in external management servers. This is a matter for each provider. While the full set of functional blocks (or their equivalent) are expected to be in place in downstream providers, MESCAL does *not* assume that *automated* processes will always implement all blocks. It is possible to deploy much of the management plane through manual processes, although this may be at the cost of reduced responsiveness or flexibility. The MESCAL functional architecture, however, shows the full set of processes that are required. For some of the service options identified in this deliverable, the algorithms or manual processes required to implement the functionality might be trivial. For instance, the *loose guarantees service option* introduced in Chapter 7 does not require explicit admission control functionality in the SLS Invocation Handling block, and the QC Mapping, Binding and Activation processes are simplified due to its adoption of well-known Meta-QoS-Classes and the restriction to bindings only with the same Meta-QoS-Class in service peer domains.

Figure 23 presents the highest-level view of the MESCAL functional architecture, showing the required functional groups.

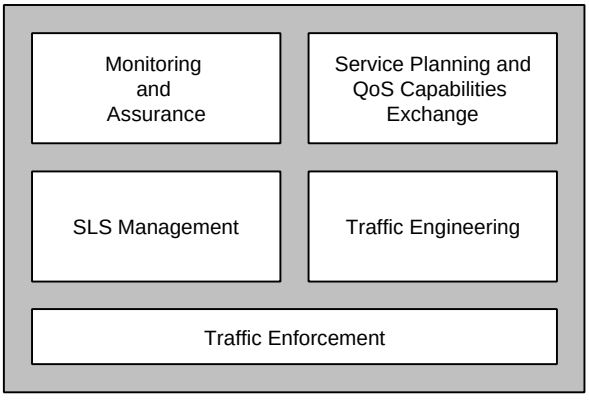


Figure 23 Abstract functional architecture

Service Planning and QoS Capabilities Exchange is responsible for defining the QoS-based services to be offered by the provider to its customers and service peers. The services are defined in terms their QoS characteristics (cf. the notion of QoS Classes in section 4.2.2), costs, capacities and destinations. These aspects are advertised to potential customers/service peers and the characteristics. In addition this functional group allows the provider to discover the services offered by its service peers,

The *Traffic Engineering* functional group is responsible for configuring and controlling the necessary resources so that the services can be delivered within its domain and across service peer networks within the contracted performance levels.

The *SLS Management* functional group negotiates contracts with customers and service peer domains based on the QoS-based services defined by *Service Planning* and implemented by *Traffic Engineering*. *SLS Management* is also responsible for controlling the admission of QoS-based service requests, undertaking authentication and authorisation tasks and ensuring that the network is not overwhelmed by traffic to the point that performance is deteriorated.

The *Monitoring and Assurance* functional group provides raw data and derived statistics to the other functional entities and measures network performance to ensure it is within contracted levels, according to agreed SLSs.

The *Traffic Enforcement* group encompasses the data plane of the provider's network. The network equipment must be configured, controlled and managed by *Traffic Engineering* and *SLS Management* so that packets are forwarded and treated according to the contracted QoS levels.

Several other functional groups could also be identified, e.g. accounting and billing, network planning, and content provision, but these are outside of the scope of MESCAL, which concentrates on the provisioning and delivery of inter-domain QoS-based services.

Figure 24 decomposes the functional groups further and maps them to the Management, Control and Data planes of network operation.

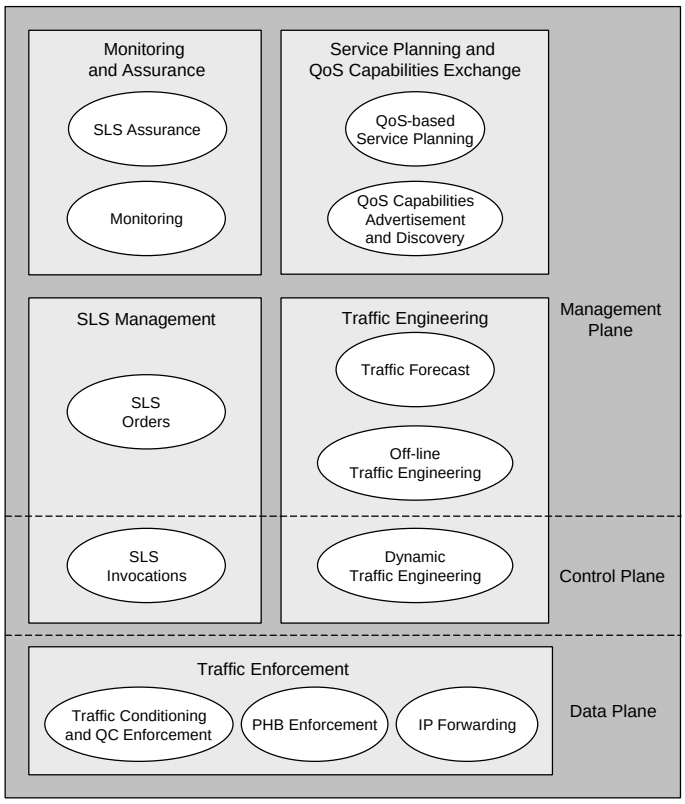


Figure 24 Intermediate decomposition of the MESCAL functional architecture

The data plane is responsible for per packet treatment within packet arrival epochs. The control plane covers intra- and inter-domain routing, SLS invocation handling – including authentication,

authorisation and admission control – dynamic resource management – including load distribution and capacity management functions. Typically, control plane functions are embedded within network equipment although they are not involved in packet-by-packet decisions.

The management plane is off-line functionality, typically located outside of the network elements in management servers. The management plane functions are responsible for planning, dimensioning and configuring the control and data planes and interacting with customers and service peers to negotiate contracts. While management plane functions are not as dynamic as control and data plane functions they are by no means static. Within the MESCAL system there is a continual background activity within the management plane at the epochs of the so-called resource provisioning cycles (RPCs). The concept of the RPC was first introduced by the TEQUILA project [Goder02a] while studying intra-domain traffic engineering for QoS-based services. There are two RPCs in MESCAL – the intra-domain RPC which involves off-line intra-domain Traffic Engineering, and the inter-domain RPC which involves off-line inter-domain Traffic Engineering. The RPCs aim at proactively optimising network resources to meet predicted demand and to build in sufficient spare capacity to avoid the burden of reconfiguring the network for each and every SLS subscription or renegotiation, without the inefficiencies and costs associated with massively over provisioning resources.

Figure 25 introduces the MESCAL functional architecture, showing the interactions between functional blocks at a high level. The arrows depict the direction of the main flow of information between functional blocks, generally implying a configuration or the invocation of a method in the direction of the arrow. The semantics of the interactions will be specified in detail in deliverable D1.2.

It should be noted that some of the functional blocks may be decomposed further, for example Dynamic Intra-domain Traffic Engineering could be split into intra-domain routing, dynamic capacity management and dynamic route management (load balancing) components. The detailed decomposition will be undertaken when the algorithms and protocols required by each block are specified in deliverable D1.2, and as the specifications are mapped to an engineering architecture for implementation.

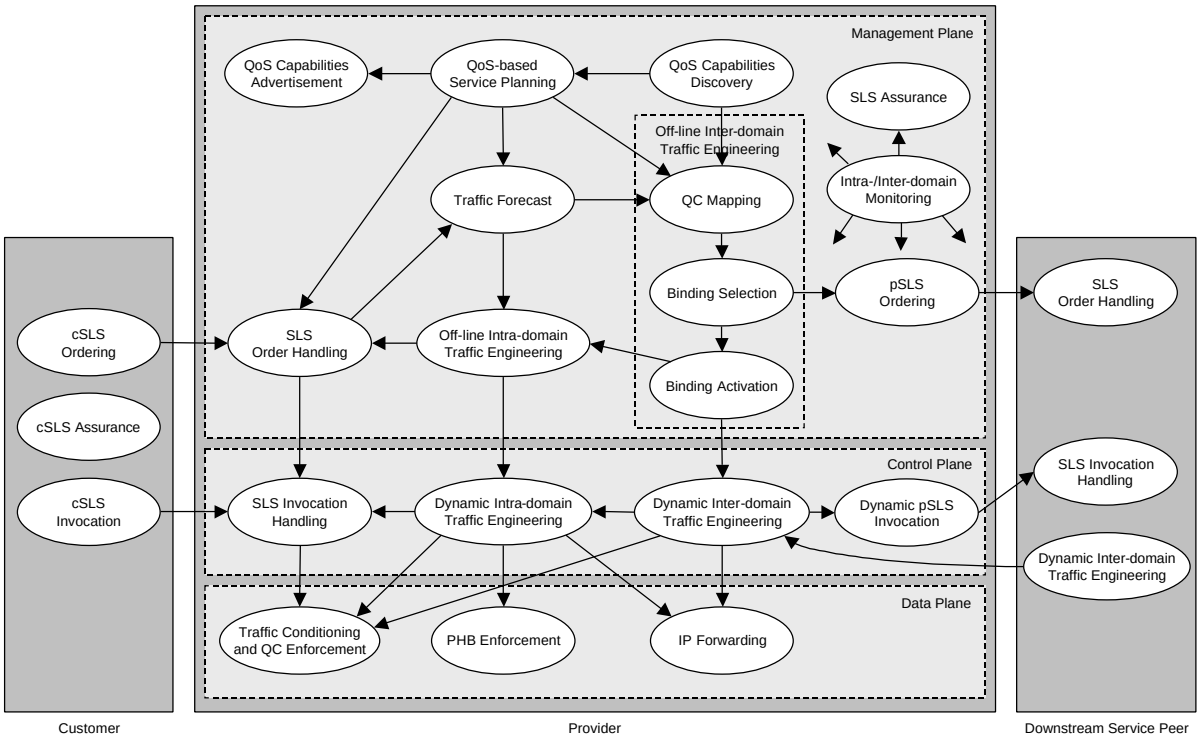


Figure 25 The MESCAL functional architecture

Figure 25 also shows the interactions between providers and between customers and providers. The downstream provider on the right of the figure shows only the components directly involved in service peer interactions.

The following subsections identify the major aspects of the functionality contained within each of the blocks shown in Figure 25.

6.2 QoS-based Service Planning

As far as the SLS Management and Traffic Engineering functions are concerned *QoS-based Service Planning* encompasses all the higher level business related activities responsible for defining the services that the provider should offer to its customers and service peer providers. These are specified according to the business objectives of the provider, and include l-QCs within the scope of its own network and e-QCs combining its local QoS-based services with those offered by its service peers. e-QCs are planned in order to extend the reach of the provider's own local QoS-based services to remote destinations or to provide its local customers access to remote services accessible through service peer providers. In addition *QoS-based Service Planning* will identify the commercial case for offering transit services to potential upstream service peer providers to access services offered by downstream service peers with which it will pre-negotiate pSLSs.

QoS-based Service Planning is made aware of the QoS capabilities of service peer providers (o-QCs) and their reachable destinations by the *QoS Capabilities Discovery* functional block. *QoS-based Service Planning* must provide the *Off-line Inter-domain Traffic Engineering* components, especially *QC Mapping*, with the target e-QCs QoS parameters, together with the required destination prefixes (or ASs) and other constraints such as cost, minimum bandwidth or allowable AS paths {although this is greatly simplified in the case of the loose guarantees service option). *SLS Order Handling* must be informed of the services the provider is able to offer as SLSs. The *QoS-based Service Planning* functional block also provides traffic demand estimates for the services it has defined to *Traffic Forecast* so that the *Intra-* and *Inter-domain Traffic Engineering* blocks may provision sufficient resources for the anticipated demand. This is especially important for new services where Traffic Forecast does not have historical information from *Monitoring* or *SLS Order Handling* to accurately predict demand. In subsequent provisioning cycles it is expected that *Traffic Forecast* will be able to predict traffic demand more accurately by observing SLS orders and actual resource usage.

6.3 QoS Capabilities Discovery and Advertisement

A provider discovers the QoS capabilities, capacities, destination prefixes and costs of service peer providers thanks to the *QoS Capabilities Discovery* functional block. Once l-QCs and e-QCs have been defined and engineered (by Intra- and/or Inter-domain TE) the *QoS Capabilities Advertisement* block is responsible for promoting the offered services so that its customers and service peer providers are aware of its offerings. It is envisaged that a variety of advertising means will be used, ranging from digital marketplaces or other automated peer-to-peer processes to conventional techniques such as salespersons, newspapers and word of mouth.

An advertisement contains details of the o-QC being offered by the domain, time schedule constraints and reachability information. The reachability information indicates the destination addresses/prefixes and/or the ASs that the domain can reach with the o-QC.

An advertisement could also contain an indication of the bandwidth and cost associated with the o-QC to enable preliminary selection when constructing potential e-QCs, although the bandwidth/cost aspects will be confirmed during pSLS negotiation.

6.4 Traffic Forecast

Traffic Forecast is responsible for aggregating and forecasting traffic demand. During a provisioning cycle, cSLSs and pSLSs are retrieved from the SLS Repository (part of the *SLS Order Handling* functional block, but not shown at this level of functional decomposition) and an aggregation process combines SLSs with the equivalent l/e-QC and ingress-egress requirements. This results in a traffic matrix with the demand per ordered aggregate between ingress and egress points of the domain (ASBRs).

Traffic Forecast uses historical records of the traffic matrix from previous provisioning cycles and actual network resource usage information from monitoring to forecast the likely growth in demand during the forthcoming provisioning epoch. This forecast will ensure that the traffic engineering processes will dimension the local and inter-domain resources (pSLSs) to accommodate established SLSs as well as those anticipated to be ordered during the current provisioning cycle.

6.5 Off-line Inter-domain Traffic Engineering

6.5.1 QC Mapping

QC Mapping is the process of combining l-QCs of the local domain with o-QCs of other domains, learned through *QoS Capabilities Discovery,* to construct potential e-QCs that meet the service requirements defined by *QoS-based Service Planning*. The combinations might be based on any grounds of compatibility deemed appropriate by the provider domain to build the e-QC e.g. based on Meta-QoS-Class equivalence or g-QC conformance criteria.

It should be noted that for an e-QC deemed necessary to be provided, a number of combinations could be potentially made. For example, this may be the case when the provider domain provides more than one l-QC for the same Meta-QoS-Class. *QC Mapping* determines a subset of the compatible combinations that could be possibly made.

QC Mapping considers the QC and the time schedule requirements of the traffic matrix, but it does not take the bandwidth and cost constraints into account. The *Binding Selection* functional block considers the latter.

The process of combining l-QCs and o-QCs differs for cascaded and centralised approaches. For a cascaded approach, it is necessary to discover o-QCs that can reach the required destination but for centralised approaches, it is necessary to additionally calculate sequences of domains to reach the destination.

6.5.2 Binding Selection

Binding Selection is responsible for selecting, from the QC mapping options, the binding of l-QCs of the local domain to the o-QCs of peer service domains. The selection uses the bandwidth and cost constraints in the traffic matrix. The latter constraints could be made available through the QoS advertisement operation. *Binding Selection* process drives *pSLS Ordering* to confirm the availability and cost of the pSLS.

It should be noted that *Binding Selection* might result in a number of QoS-bindings for a given e-QC. QoS-bindings with the same service-peering provider may differ in the l-QC and subsequently in the o-QC they use. Alternatively, QoS-bindings may differ when established with different service-peering providers. Providers may find such multiplicity advantageous for avoiding to be bound to a specific QoS-capability of a particular service-peering provider and/or exploit the merits of dynamic, multi-path routing –note that different bindings imply different intra- and inter-domain routes in general.

6.5.3 Binding Activation

Binding Activation is an offline component that runs at inter-domain Resource Provisioning Cycle epochs. *Binding Activation* is responsible for mapping the predicted traffic matrix to the inter-domain network resources (once pSLSs have been established), satisfying QoS requirements while aiming at optimising the use of network resources across AS boundaries. *Binding Activation* decides which of the established QoS-bindings will be *put in effect* in the network for implementing an e-QC together with the associated routing constraints for those e-QCs. A provider domain may decide to put in effect only one of the determined bindings at a time, switching to another binding under appropriate conditions. Alternatively, a provider domain may decide to put in effect all determined bindings and employ a dynamic routing scheme to select between them. This involves producing directives for

inter-domain routing to define multiple AS paths together with the initial values of the traffic splitting ratio for load balancing. The QC-bindings in effect will be enforced through routing decisions as well as configurations of the *Traffic Conditioning and QC Enforcement* block, e.g. configuring the egress ASBR to perform DSCP remarking for realising a QC-binding. The latter configuration can be made directly to the egress router or passed through *Dynamic Inter-domain Traffic Engineering*.

6.6 Dynamic Inter-domain Traffic Engineering

Dynamic Inter-domain Traffic Engineering runs within an inter-domain RPC and is responsible for inter-domain routing e.g. qBGP advertisement, qBGP path selection and for dynamically performing load balancing between the multiple paths defined by the static component based on real-time monitoring information changing appropriately the ratio of the traffic mapped on to the inter-domain paths. This component may pass the required information to *Dynamic Intra-domain Traffic Engineering* or it may directly configure the appropriate load balancing mechanism in the *IP Forwarding* block.

6.7 SLS Order Handling

SLS Order Handling is the functional block implementing the server side of the SLS negotiation process. Its job is to perform subscription level admission control. The *Off-line Intra-domain Traffic Engineering* block will provide *SLS Order Handling* with the resource availability matrix (RAM) which indicates the available capacity of the engineered network to accept new SLS orders – both within the AS and on any inter-domain pSLSs it has with neighbouring ASs. *SLS Order Handling* will negotiate the subscription of both cSLSs and pSLSs – they will be (largely) treated in the same way. *SLS Order Handling* maps incoming SLS requests onto the o-QCs it can offer and investigate whether there is sufficient intra- and inter-domain capacity, based on the RAM for that o-QC. There will be a certain amount of overbooking allowed, depending on policies set by *Policy Management* (not shown in the functional architecture, but see section 6.19). Successfully negotiated SLSs are stored in the SLS repository (part of *SLS Order Handling*, but not shown at this level of decomposition) and *SLS Invocation Handling* is configured appropriately to allow future invocations on the new SLS. The contents of the SLS repository are used as an input to *Traffic Forecast* for future resource provisioning cycles. If there is insufficient capacity – either within the AS or on the pSLS with peer ASs – then the negotiation will fail. When *SLS Order Handling* sees that the RAM minus the new SLSs since the last RPC is small (as defined by a policy) it will trigger a new RPC.

6.8 pSLS Ordering

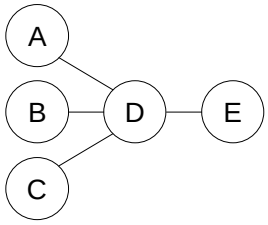
pSLS Ordering is the client side of the pSLS negotiation process, which interacts with a *SLS Order Handling* block in the service peer. During an inter-domain RPC *Binding Selection* may identify the need for new pSLSs with service peers or for existing pSLS to be re-negotiated. In the latter case this will usually be limited to bandwidth modifications. *pSLS Ordering* implements the decisions of the *Binding Selection* algorithms and undertakes the negotiation process. *Binding Selection* will identify the upper and lower bounds of the desired SLS parameters (e.g. delay, cost) or cost/performance ratios to guide the negotiation process and the limits beyond which an offered pSLS would be unacceptable. There may be a need for a transaction orientated negotiation to cope with the situation when several pSLSs need to be negotiated in parallel, perhaps with different ASs (e.g. if there is load to be split between two downstream ASs, for the same e-QC and destination), if one should fail then they should all fail (if there would not be sufficient capacity on a single pSLS for the aggregate traffic, in the same example).

6.9 Dynamic pSLS Invocation

This block is the client side of the pSLS invocation procedure. pSLSs are always on (unlike many cSLSs which need to be explicitly invoked for each flow) because they are supporting aggregate traffic from many cSLS flows. However the aggregate rate of the traffic may fluctuate significantly

within RPCs. The *Dynamic pSLS Invocation* functional block will monitor the usage of the pSLS from the client's perspective and by extrapolating trend information it can estimate future requirements and signal requests to increase or reduce pSLS capacity. Any increase or reduction in capacity should always be within the range allowed by the terms of the subscribed pSLS. If this block determines that the pSLS should be outside of the agreed limits then it should notify the *pSLS Ordering* block (either directly or indirectly through *Monitoring*, and *Traffic Forecast*) so that the pSLS contract may be renegotiated during the next RPC.

The *SLS Order Handling* block can overbook resources provided that *SLS Invocation Handling* ensures that invocations are blocked during times of contention – to avoid congestion and QoS deterioration. Imagine 5 ASs in the figure below. A, B and C all have pSLSs with D to destinations in AS E. AS D multiplexes the traffic from A, B and C over a single pSLS with E. AS D may overbook/statistically multiplex traffic from A, B and C on its SLS to E provided that it has the means to ensure that performance is not deteriorated. By using an admission control scheme within the SLS Invocation Handling block for pSLS invocations, AS D can achieve this.



This is to the benefit of both upstream and downstream ASs. Downstream providers benefit from overbooking, some benefits may be passed on to the upstream providers as cheaper pSLSs. Upstream providers can invoke just the amount of capacity they need, and furthermore this scheme may allow them to occasionally exceed their subscribed capacities as the downstream AS will know whether there is sufficient spare capacity.

6.10 SLS Invocation Handling

As explained above, admission control is needed to ensure that the network is not overwhelmed with traffic when the network adopts a policy of overbooking network resources at the subscription level. *SLS Invocation Handling*, containing the admission control algorithm, receives signalling requests from customers or peer providers for cSLS and pSLS invocations respectively. *SLS Invocation Handling* checks whether the invocation is conformant to the subscribed SLS and whether there is sufficient capacity in the local AS and also on the inter-domain pSLSs in the case of SLSs that are not terminated locally. There are various approaches to admission control, including model and measurement based. If there is capacity from the *requested invocation SLS Invocation Handling configures the Traffic* Conditioning and QC Enforcement block appropriately so that packets may be forwarded accordingly.

6.11 Intra-/Inter-domain Monitoring

Monitoring is responsible for both node and network level monitoring through both passive and active techniques. It is able to collect data at the request of the other functional blocks and asynchronously notify the other functional blocks when thresholds are crossed on both elementary data and derived statistics.

For simplicity in the diagram the full set of interactions with *Monitoring* is not depicted, however *SLS Invocation Handling*, *Dynamic Inter-/Intra-domain Traffic Engineering* and *Dynamic pSLS Invocation* blocks continually use monitored data in order to operate. The less dynamic *Off-line Inter-/Intra-domain Traffic Engineering* functions as well as *Traffic Forecast* use monitored network statistics at RPC epochs. *Traffic Forecast* uses historical data to improve the accuracy of future traffic matrix estimates.

Inter-domain monitoring could take several forms: monitoring inter-domain links (pSLS) only; monitoring end-to-end performance across several ASs through loop-backs or remote probes for one-way measurements; collection of data generated by service peers (possibly through BGP advertisements, or through another monitoring data exchange protocol). Alternatively third part auditing may be a more acceptable means for both monitored and monitoring ASs.

Monitoring will not be studied in detail in MESCAL – SCAMPI, INTERMON and other projects are investigating these topics in more depth than could be done in MESCAL.

6.12 SLS Assurance

SLS Assurance compares monitored performance statistics to the contracted QoS levels agreed in the SLSs to confirm that the network or service peer-networks are delivering the agreed service levels.

6.13 Off-line Intra-domain Traffic Engineering

This component performs the main task of intra-domain TE at intra-domain RPC epochs running an off-line algorithm that computes the intra-domain network configuration in terms of routing constraints and PHB capacity requirements in order to satisfy the predicted traffic demand. The decisions of the *Off-line **Inter**-domain Traffic Engineering* should also be taken into account. The configuration is passed directly to the routers or via the *Dynamic Intra-domain TE* component. This component returns an enhanced Resource Availability Matrix (compared to that defined in the TEQUILA project [TEQUI]) with the inter-domain available resources in order for the *SLS Order Handling* block to perform subscription level "admission control" on pSLS, cSLS requests.

6.14 Dynamic Intra-domain Traffic Engineering

This is basically the dynamic management layer as defined in TEQUILA [TEQUI] which intends to manage the resources allocated by *Off-line Intra-domain Traffic Engineering* during the system operation in real-time, in order to react to statistical traffic fluctuations and special arising conditions within an intra-domain RPC. It basically monitors the network resources and is responsible for managing the routing processes dynamically as well ensuring that the capacity is appropriately distributed among the PHBs. This block includes also QoS-aware intra-domain routing e.g. CSPF.

6.15 Traffic Conditioning and QC Enforcement

Traffic Conditioning and QC Enforcement is responsible for packet classification, policing, traffic shaping and DSCP marking according to the conditions laid out in previously agreed SLSs and the invocation of those SLSs. At ingress routers the *Traffic Conditioning* function is responsible for classifying incoming packets to their o-QC and subsequently mark them with the correct DSCP for the required l-QC. At the egress router the *QC Enforcement* function may need to remark outgoing packets with the correct DSCP as agreed in the pSLS with the service peer. In other words *QC Enforcement* is responsible for implementing the binding from l-QC to o-QC of the service peer. Note that *QC Enforcement* is not responsible for selecting the correct peer AS: this is decided by routing (part of the *Dynamic Traffic Engineering* blocks in Figure 25), therefore *QC Enforcement* does not implement the full QC mapping/binding process in the data plane.

6.16 PHB Enforcement

This block represents the queuing and scheduling mechanisms required to be present in order to realise the different PHBs with the appropriate configuration as defined by the TE related blocks.

6.17 IP Forwarding

This block represents the functionality needed to forward IP datagrams based on the information maintained in the corresponding FIBs. Optionally, IP forwarding may also include mechanisms to perform multipath load balancing.

6.18 Note on Load balancing

In MESCAL there are 3 different kinds of load balancing: a) load balancing between different inter-AS routes (to different egresses) b) load balancing between different intra-domain routes (to the same egress) and c) QC load balancing (it can be both to the same egress and to different ones. The first type is included in the Inter-TE blocks, the second in the intra-TE blocks while the third belongs both in inter- and intra- TE (also see section 5.4.2).

6.19 Other functions and capabilities

The functional architecture covers those capabilities necessary for deploying and operating inter-domain QoS services. A provider may need other more general support functions such as network planning, fault and configuration management, but as these are not an explicit part of the inter-domain QoS provision problem they are not covered in this architecture. Where some aspects of these functions are required for developing prototypes and conducting experimental work the essential functionality will be investigated during the engineering design phase of the MESCAL system.

It is envisioned that rather than being entirely hard-coded at development or installation time, the behaviour of many of the MESCAL functions and algorithms can be influenced at run time by a *Policy Management* infrastructure. Policies are expected to cover the *SLS Management* and *Traffic Engineering* functional blocks. There are no explicit functional blocks to handle multicast services. It is assumed that this is distributed over many of the blocks e.g. SLS Order Handling for multicast related SLSs.

For most providers, an important aspect of providing service differentiation is the means for charging appropriate rates for different services levels. Metering, rating, billing and other commercial aspects of QoS delivery are outside of the scope of MESCAL and are therefore not part of the specified functionality, which is focussed on the technical capabilities comprising the main open areas of research. Because MESCAL has adopted a cascaded model of inter-domain services it is foreseen that end-to-end charging models are scalable as they do not require explicit charging in each provider domain for every cSLS invocation or QoS flow. Inter-domain accounting can be done at the level of QoS aggregates and charging/billing is only an issue between directly attached service peers/customers. Provided that SLS orders and invocations can be tracked and, in the worst case, the DSCP of packets can be traced to provide QoS-class metering, existing accounting practices can be adapted for inter-domain QoS-based charging. MESCAL does not aim to provide accounting solutions, which are being studied by others, but the project will raise any additional requirements on inter-domain accounting that arise during its theoretical, specification and prototyping activities.

7 SOLUTION SPACE

7.1 Introduction

Based on the customer requirements listed in section 3.2.3, the MESCAL project has identified, at a high level, two major end-users categories. These categories differ at the level of the QoS guarantees they require, the topological scope of their SLSs and by the permanence of their communications requirements.

Residential customers may subscribe to IP services such as VoIP, video on demand and broadcasting services. These users may want to reach any available destination at any time without being tied to a single destination, or limited set of destinations at subscription time. The duration of the communications between one of these end-users and a *specific* content provider or peer customer may be short (just the duration of a service transaction for instance) and the frequency of interactions can be sparse. In the case of peer-to-peer file sharing applications or premium web-browsing, for example, the total sum of the communications requirements from one customer to a large number of destinations may be relatively long lived with a dense frequency of interactions.

On the other hand, *corporate* customers, may request specific, strong guarantees for supporting particular mission- or safety-critical applications and services, such as IP VPNs, Virtual Leased Lines, corporate VoIP services, remote control of equipment such as control of robot arms or surgical instruments. These requirements are usually to a limited, small set of destinations, the relationship between the communicating entities is long-lived and the frequency of interactions is usually dense.

These two categories can be seen as two extremes: the residential customer wants to communicate with all destinations with better-than-best-effort service levels, while the corporate customer wants a point-to-point pipe to a named destination with hard upper bounds on QoS and a constant bandwidth. Obviously these are two extreme cases and a range of customer categories could be identified between these two, such as the customer requiring hard upper bounds on delay to a large but limited set of destinations with statistically guaranteed throughput.

From a contractual viewpoint these requirements introduce some variations in the way the following SLS parameters are handled:

- Topological scope: which is "any" for residential customers but is usually a limited set of specific destinations for corporate business customers.
- End-to-end QoS guarantees: residential customers may have only loose requirements which could be captured in qualitative parameters while corporate customers may require explicit hard guarantees with specific values for the upper bounds on loss, delay and jitter, for example.
- End-to-end bandwidth guarantees: corporate customers require at least a statistical guarantee, if not a hard peak-rate allocation, of the bandwidth specified in its SLSs. Residential customers may be content with best effort bandwidth availability or may require some statistical guarantees, but they are unlikely to be willing to pay the premiums associated with peak rate end-to-end bandwidth reservations.

It is intuitively obvious that end-to-end hard QoS performance and bandwidth guarantees cannot be offered to all Internet users with the level of dynamics that characterises the large number of residential customers. This is mainly due to scalability reasons: IntServ was widely seen as unscalable even *within* domains, for example. In order satisfy the requirements of the aforementioned customer categories MESCAL has specified a solution space encompassing three main service options.

These service options are discussed in 7.2. Note that a given provider could support all or only a subset of these service options. In section 7.3 we provide the details of the MESCAL solution, evaluate its conformance against the provider and customer requirements and map it to the MESCAL functional architecture.

7.2 Service Options

Previous chapters have described the Inter-domain QoS requirements that the MESCAL solution must meet, from both provider and customer perspectives. MESCAL has identified three service options characterised by the level of guarantee they can provide:

- The *Loose Guarantees* service option, which globally aims at providing better Internet-based services, but doesn't provide any strong guarantees.
- The *Statistical Guarantees* service option, which offers QoS performance guarantees for specific destinations and which allows some loose end-to-end bandwidth guarantees.
- The *Hard Guarantees* service option, which improves the above option with strong end-to-end bandwidth guarantees.

These service options provide distinct and different service characteristics, which enable providers to meet the requirements of a diverse range of customers, see Table 2, below.

	Service Options		
Characteristics	Loose	Statistical	Hard
E2E QoS Performance	Qualitative	Qualitative/Quantitative (statistical guarantee)	Quantitative
E2E Bandwidth	No guarantee	Statistical guarantee	Guaranteed
Topological Scope	Any reachable destination	Specific destinations	Specific destinations

Table 2: MESCAL Service Options

The MESCAL Loose service option enables a provider to offer customers access to differentiated transport services, where each differentiated service is related to a Meta-QoS-Class. It is envisaged that providers throughout the Internet will implement a small number of well-known Meta-QoS-Classes. Inter-domain QoS services are then created by constructing paths across those domains that support a particular Meta-QoS-Class. In effect, a set of parallel “internets” are deployed, each offering service levels associated with a specific Meta-QoS-Class. The guarantees associated with the Loose service are restricted to qualitative services, although it is anticipated that the characteristics of each Meta-QoS-Class based service will be based on common application requirements, for example VoIP. The Loose service option does not provide any end-to-end bandwidth guarantees because the option enables any destination to be reached, without prior identification in the cSLS/pSLS. The objective of the Loose service option is to address the requirements of a large population of users, while keeping the network engineering as simple as possible by supporting relaxed service guarantees.

The MESCAL Statistical service option provides customers access to inter-domain QoS services with firmer guarantees than the Loose option. The Statistical service option is able to provide a qualitative QoS service, although quantitative services where values for packet delay and loss are specified can also be offered. Additionally, an end-to-end bandwidth guarantee is provided within statistical bounds. An Inter-domain QoS service based on the MESCAL Statistical option is created by constructing paths across domains that are able to guarantee their QoS capabilities. QoS services can either be constructed to meet specific quantified QoS constraints or the Meta-QoS-Class approach can be used for offering qualitative services. A distinguishing feature of this service option is that the guarantees are statistical. It is a policy decision for each provider to decide the level of the guarantee that it wants to offer and it is to be expected that QoS services with firmer guarantees will require higher allocation of resources in the provider’s network.

The MESCAL Hard service option provides customers with strict inter-domain performance guarantees. The Hard service option is targeted at providing services with quantitative QoS and bandwidth guarantees with a high probability of fulfilment. An Inter-domain QoS service based on the MESCAL Hard option is created by constructing paths across domains that are able to guarantee their

QoS capabilities to the required level. It is envisaged that network resources will have to be permanently allocated for this service and consequently, the MESCAL Hard service option is suitable for services that can justify the high costs that will be associated with the service. The Hard service option will be appropriate for a small number of added-value services, such as critical business services.

7.3 The MESCAL Solution

The purpose of this section is to describe the MESCAL solution to supporting the three identified service options. The MESCAL solution is directly mapped to the Functional Architecture, see section 6, for each of the service options and is conformant with both the customer and provider requirements, which have been identified in section 3.2. We provide the detailed description of all the required QC-operations in order to have as a result the required infrastructure to achieve the objectives of each of the service options.

Based on the service options described above, the MESCAL project has identified three solution options that target three different end-users categories:

- The Loose Guarantees solution option: this solution option aims at providing an implementation of the Loose Guarantees service option that has been described above. This option allows having some QoS treatment when this is possible. No strict guarantees are assumed by this option.
- The Statistical Guarantees solution option: this solution option is based on the statistic service option.
- The Hard Guarantees solution option: this solution option gives hard guarantees to the customers.

7.3.1 Loose Guarantees Solution Option

This solution option aims at providing an implementation of the Loose Guarantees service option, which has been introduced above.

A light version of this solution option is also presented. This version changes the way the mapping operation is done and makes the signalling operation less heavy than the non-light version.

7.3.1.1 Use of Meta-QoS-Class concept

The underlying philosophy relies on the assumption that wherever end-users are connected they use similar applications in similar business contexts. They also experience the same QoS difficulties and are lead to express similar QoS requirements to their respective service providers.

The solution to this service option assumes that providers will define and deploy similar classes of service because they are in general confronted with the same customers requirements. These classes target to support applications, which have similar QoS constraints. There are no reasons to consider that a provider in Japan would design a "Voice Over IP" l-QC with short delay, low loss and small jitter while another one in Germany would have a completely different view. Constraints are implicitly imposed by applications to the network, independently of where the service is used or accessed.

It should be clear that a *Meta-QoS-Class* is actually an abstract concept. It is not a real l-QC provisioned in a real network. A *Meta-QoS-Class* is defined in terms of services (e.g. VoIP) and can be given some boundaries for QoS performance attributes. This point is a fundamental aspect of this solution option.

For instance, *Meta-QoS-Classes* could inherit from each other as follows:

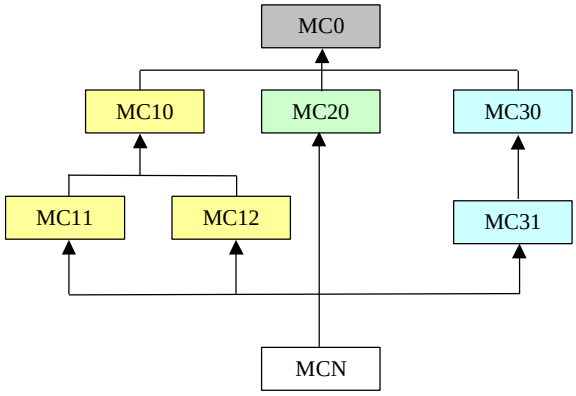


Figure 26: *Meta-QoS-Class* inheritance example diagram

In this example, MC0 represents the BE effort *Meta-QoS-Class* and MCN is the "impossible to get" neutral element. Each branch of the tree is designed to be suitable for different QoS application. The categories of applications are generic and are described in network performance terms (mainly in terms of sensitivity to delay, jitter, loss or any other network performance characteristic which can be qualitative and/or quantitative).

If several grades of QoS are considered for an application category, *Meta-QoS-Classes* can be defined to form a hierarchical tree. In this particular example, this means that MC11 would also be suitable for conveying flows requesting MC10 and MCN could potentially be used for any kind of traffic since it represents the neutral element. This hierarchical ordering of *Meta-QoS-Class* is an assumption and, at this stage, it is still uncertain whether branch splitting (MC11 and MC12 for instance) should be conceptually kept in future specification.

7.3.1.2 QC-classification

Before involved in any other inter-domain QoS related operation, each provider must classify his l-QCs with regard to *Meta-QoS-Classes*. This is what is named the *QC-classification* process. This operation occurs each time a new l-QC is designed or an existing one re-engineered. An l-QC can potentially satisfy several *Meta-QoS-Classes.*

For instance, a provider could have defined:

- `l-QC20`: satisfies MC0, noted `l-QC20[MC0]`,
- `l-QC21`: satisfies MC10 and MC20, noted l-QC`21[MC10,MC20]`
- `l-QC22`: satisfies MC11, noted l-QC`22[MC11]`.

In the *Light approach*, an l-QC can satisfy one and only one *Meta-QoS-Class*. *Meta-QoS-Classes* inheritance properties cannot be used. *Meta-QoS-Class* concept is only used for mapping and binding purpose.

7.3.1.3 QC-mapping

From a business perspective, a provider can logically express the need to extend its own classes of service across the Internet. In particular, this means that a flow originated in the provider's AS, with an indication of the requested class of service, should experience a similar treatment when crossing the set of various autonomous systems up to its final destination. For doing that, the provider must establish peering contracts (pSLSs).

But first of all, before the establishment of any *pSLS*, the provider requesting the pSLS must proceed to a *QC-mapping* in order to identify the whole set of potentially compatible bindings between its own l-QCs and the remote's o-QCs with the objective to extend the scope of its services over the Internet.

The solution to the loose guarantees service option defines that the QC-mapping concerns only the *Meta-QoS-Classes* that the provider decides to extend. This compatibility-mapping criterion is ensured by the *Meta-QoS-Class* concept. Two classes are declared to be compatible for mapping if they belong to the same *Meta-QoS-Class*, directly or by inheritance.

For achieving this QC-mapping the service peer provider AS must indicate to the requestor AS if it supports each of the requested *Meta-QoS-Classes*.

In the *Light approach*, the requesting provider will consider all possible mappings between each of its l-QCs **with only one** of the remote o-QC providing that the remote o-QC belongs to either the same or a better *Meta-QoS-Class*.

7.3.1.4 QC-binding

In the context of the MESCAL solution for supporting the loose guarantees service option, QC-binding concerns only the *Meta-QoS-Classes* the requesting AS decides to extend. The QC-binding process becomes very simple and can be summarised as a binary assessment: *does the peering partner support the requested Meta-QoS-Class or not?* All the same, there can be a very limited number of combinations if the service peering provider gives a choice of the l-QC it will use to implement the Meta-QoS-Class but that's all.

At the end of this process, several l-QCs from the requesting AS can be potentially used for transporting datagrams that belong to the same *Meta-QoS-Class*. On its side, the peering provider can choose to select only one or several of its compatible l-QCs to fulfil the contractual terms of the pSLS.

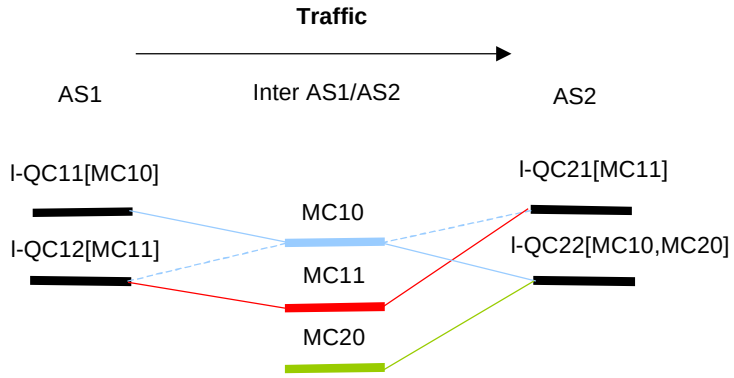


Figure 27: Example of the QC-binding operation

In Figure 27 we show an example of the QC-binding operation. Within AS1, as a result of the QC-classification operation, the MC10 traffic can be assigned to l-QC11 or l-QC12. The MC11 traffic can only be transported with l-QC12. MC20 is not supported by AS1. In AS2, as a result of the QC-classification operation, the MC10 traffic can be assigned to l-QC21 or l-QC22. The MC11 traffic can only be assigned to l-QC21. MC20 is transported by QC22.

In the above example, at the highest level, the QC-binding lead AS1 to exchange MC10 and MC11 traffic. In detailed the following binding have been achieved:

- l-QC11 -> l-QC21 or l-QC22
- l-QC12 -> l-QC21 or l-QC22

Depending on the *Meta-QoS-Class*, one of the 4 possible bindings is used.

In the *Light approach* the same figure would become:

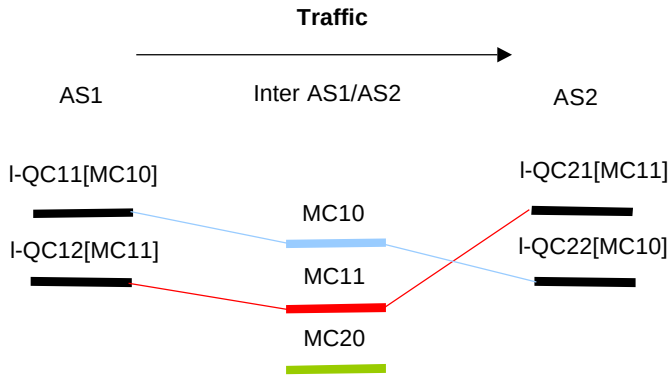


Figure 28: Example of the QC-binding operation with the Light approach

In AS1, as a result of the QC-classification operation, the MC10 traffic can be assigned to l-QC11. MC11 traffic can only be assigned to l-QC12. MC20 is not supported by AS1.

In AS2, as a result of the QC-classification operation, the MC10 traffic can be assigned to QC22. MC11 traffic can only be transported with QC21. MC20 is not supported by AS2.

In the above example, the QC-binding lead AS1 to exchange MC10 and MC11 traffic. In detailed the following binding have been achieved:

- l-QC11 -> l-QC21
- l-QC12 -> l-QC22

7.3.1.5 QC-implementation

7.3.1.5.1 QC-Indication

An important aspect of this approach is that *Meta-QoS-Classes* are used to indicate the requested QoS across the Internet. A *Meta-QoS-Class* indicator is used both intra-domain and inter-domain. This could be a global value agreed by all providers or a local value understandable by two adjacent eBGP peers. The DSCP can be used for this purpose with the limitation of 64 values.

In intra-domain, the end-user submits a datagram with an indication of the requested *Meta-QoS-Class*. The first provider's router chooses an appropriate l-QC for transporting this datagram within the domain (since several l-QCs can potentially satisfy the same Meta-QoS-Class). This l-QC is used cross the domain and the QoS of service experienced by this datagram is compliant with that l-QC. Nevertheless, the *Meta-QoS-Class* indicator is kept in the datagram.

When the datagram reaches a domain boundary, the l-QC indicator cannot be used anymore in the remote domain and the *Meta-QoS-Class* indicator is used instead. The receiving provider then uses its own l-QC to transport the datagram up to its border router in the domain. Using a Meta-QoS-Class indication allows splitting an l-QC while avoiding the QC-splitting problem.

In the *Light approach,* there is no *Meta-QoS-Class* signalling indicator. The end-user submits a datagram using an l-QC indicator. The egress AS is supposed to indicate the remote ingress l-QC that will be used by the ingress AS, thanks to the DSCP field of the IP datagram. By definition of the mapping and splitting processes, there is no possible QC-splitting.

It should be noted that *Meta-QoS-Class* indication allows outclassing traffic (i.e. treat the traffic within a better MC) when crossing an external domain because the Meta-QoS-Class indicator is transported end-to-end by the datagram. When exiting the remote domain, the datagram can be transported by a more appropriate remote l-QC, as originally requested by the end-user.

In the *Light approach*, outclassing is also supported but once a datagram has been outclassed it cannot go back to its originally requested *Meta-QoS-Class* since the datagram doesn't convey such indicator.

In Figure 29, l-QCij have been classified as follow:

- AS1: l-QC11[MC10], l-QC12[MC11]
- AS2: l-QC21[MC11], l-QC22[MC10, MC20]
- AS3: l-QC31[MC11], l-QC32[M20], l-QC33[MC10]

In order to keep the figure simple, *Meta-QoS-Class* MC0 is not shown.

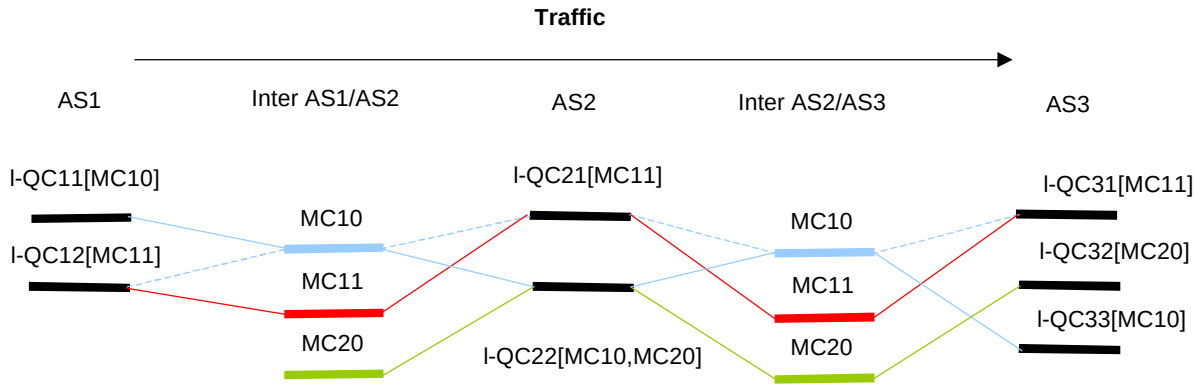


Figure 29: QC bindings in the name of *Meta-QoS-Classes*

Considering in AS1 an IP datagram marked with l-QC11 in the name of *Meta-QoS-Class* MC10 (hereafter noted l-QCij{MCx}), l-QC11 can be bound to l-QC21 or l-QC22 since those two classes are respectively mapped to MC11 (which inherits from MC10 in this example) and MC10. l-QC22 binding is probably the optimal binding for MC10 but l-QC21 is valid too. The choice to use l-QC21 in AS2 outclasses the traffic sent by AS1 for *Meta-QoS-Class* MC10. Outclassed bindings have been indicated with dotted lines.

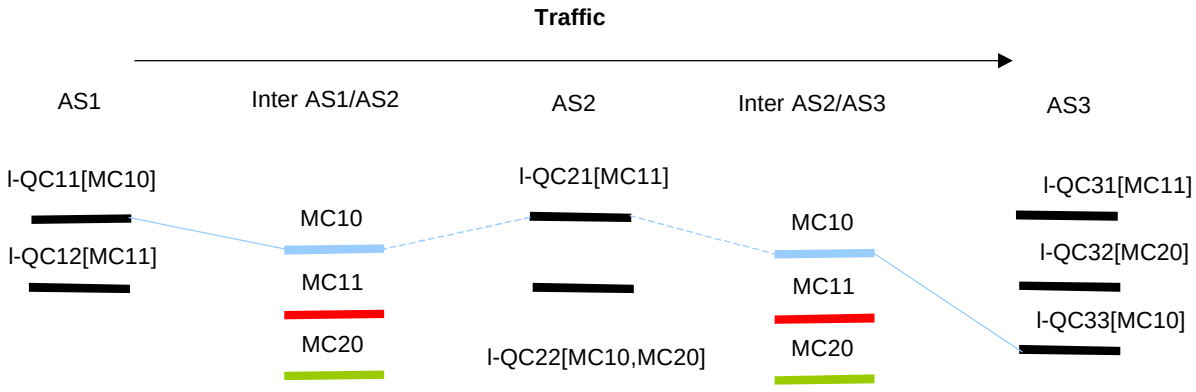


Figure 30: Temporarily outclassing example

In Figure 31, AS1 has three internal l-QCs. One of them, l-QC11, has been declared (l-QC-classification operation) as a member of a particular *Meta-QoS-Class*. In the name of this *Meta-QoS-Class*, QC bindings have been achieved iteratively across all ASes. All ASes have gone through the same process, no matter the order in which the bindings have been established. The resulting "l-QC" bindings, for this particular *Meta-QoS-Class* are depicted in red (bold for black and white restitution support).

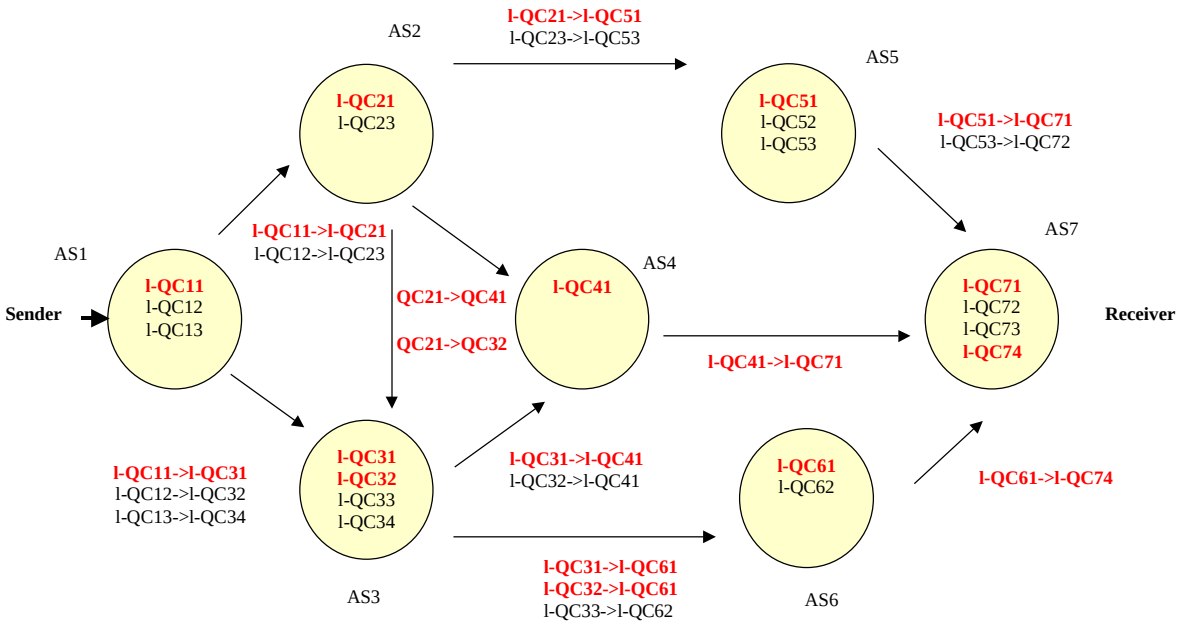


Figure 31: Following QC11 through contractual cross binding

From AS1 perspective, l-QC11 has been extended throughout the whole topology. Any sender from AS1 can reach any receiver anywhere through l-QC11 extension. At this stage, there are several possible paths from the sender to the receiver following l-QC11 extension. We'll see in the paragraph "Intra-domain and inter-domain routing aspects" how we propose to select only one path.

Figure 31 shows a connected topology. This solution option is interesting only if these bindings become common practice, so that each provider can see its own l-QCs extended throughout almost the whole Internet. However, we may reasonably expect some holes even if this solution option is largely and globally spread. The figure shows unidirectional bindings but it should be possible to establish bi-directional bindings.

7.3.1.5.2 Intra-domain and inter-domain routing aspects

7.3.1.5.2.1 Inter-domain routing: path selection

In this approach, the Internet appears as a set of parallel *Meta-QoS-Class* planes. Each *Meta-QoS-Class* plane consists of all the l-QCs bound in the name of the same *Meta-QoS-Class*. When an l-QC maps different *Meta-QoS-Classes* then it belongs to all the different *Meta-QoS-Class* planes.

We assume that in a *Meta-QoS-Class* plane, all paths are, to a reasonable extent, treated equally. Therefore, the problem of path selection amounts to: do your best to find one path for each *Meta-QoS-Class.* We rely on a BGP-like protocol for the path selection process. We call this protocol qBGP, this protocol selects and advertises one path for each *Meta-QoS-Class* plane per destination.

When, for a given *Meta-QoS-Class* plane, there is no path available to a destination, the only way for a datagram to travel to this destination is to use another *Meta-QoS-Class* plane from start. The only *Meta-QoS-Class* plane available for all destinations is the best-effort *Meta-QoS-Class* plane (also known as "the Internet"). There's no straightforward solution to change from one plane to another on the fly. So, there's no straightforward way to span a *Meta-QoS-Class* plane hole by a best-effort bridge.

When a datagram enters an AS, the AS must know in which *Meta-QoS-Class* plane it belongs to in order to retrieve the egress point selected by qBGP and also to apply the correct l-QC. QC-indication as described in 7.3.1.5.1 applies here: each datagram should convey an indicator of the *Meta-QoS-Class* it refers to.

7.3.1.5.2.2 Intra-domain routing: from the AS ingress point to the AS egress point

The intra-domain routing should also take into account the *Meta-QoS-Class* concept.

In a domain, each router will have to maintain one routing plane per *Meta-QoS-Class*. Indeed, since an l-QC can belong to several *Meta-QoS-Classes*, the same l-QC may be used for transporting traffic on behalf of different *Meta-QoS-Classes*. Egress points, for a same destination but for different *Meta-QoS-Classes,* may be different even if the same l-QC is used for crossing the domain up to the egress point. Intra-domain routing must be achieved on the destination and the *Meta-QoS-Class* indicator.

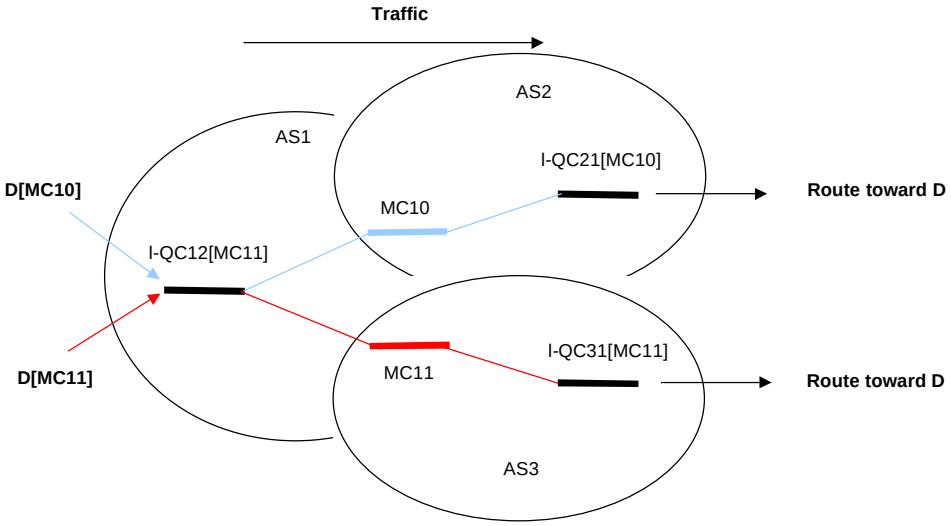


Figure 32: Example of an l-QC belonging to several Meta-QoS-Class

In the above example, two sources want to reach destination "D", somewhere in the Internet. D is not located nor in AS3 nor in AS2. One source (blue one) uses the Meta-QoS-Class MC10 and the other the MC11 (red one). They use l-QC12 to cross the first domain but the egress point of the domain is different. Routing cannot rely only on the destination address but on both the destination address and the MC. When an end-user or an external peering provider injects some traffic in the domain, the first provider router is responsible for selecting the l-QC to use for reaching the egress point. The chosen l-QC must support the requested *Meta-QoS-Class*.

In the *Light approach*, an l-QC can belong to only one *Meta-QoS-Class*. When a border router learns a destination "D" (because a pSLS exists) on behalf of QC-binding enforced, the qBGP path selection process selects the most appropriate egress point and made it known to the intra-domain routing within the l-QC routing plane concerned with this the QC-binding. If the domain binds several l-QC on the same remote l-QC, the learned destination is flooded into the corresponding l-QC planes. Consequently, there is one routing plane per l-QC and routing must take into account both the l-QC and the destination address.

7.3.1.6 IPv6 support

This approach does not use any specific IPv4 capabilities other than the DSCP field in order to signal the l-QC to use. If this solution option can be implemented using IPv4 it should also be supported by an IPv6 infrastructure.

Since it might be easier to code the MC indicator in one of the IPv6 options, this protocol may be convenient for use by MESCAL.

7.3.1.7 QoS Guarantees

In this basic approach, QoS is achieved thanks to a cascade of pSLS. If previously established pSLS are cancelled, any cSLS relying on those contracts becomes invalid. Network accessibility cannot be

ensured, and "holes" can appear anywhere at any time. The provider who establishes a cSLS cannot constrain a remote provider to maintain pSLS for its own needs.

QoS, which is experienced by the end-user, can be variable. In fact, the inter-domain route selection can change at any time: when a pSLS is cancelled, when a route is no more accessible (link down for instance), etc. In those cases, the path changes (if there is more than one) and the new QoS value can be different from the previous one.

Nevertheless, we can know, at any time, the QoS value for a given destination if we add a reporting functionality to the qBGP protocol. This mechanism would compute, in a step-by-step process, the QoS attribute `{D, J, L}` attached to each AS path and advertise it. When my AS receives from one of its peers the announcement: *`Meta-QoS-Class`* `+ (AS path) + (QoS value) + destination`, and selects this path, it advertises: *`Meta-QoS-Class`* `+ {myAS,AS path} + (myQoS ⊗ QoS value) + destination`.

Bandwidth guarantees cannot be supported.

7.3.1.8 Scalability

When qBGP is employed the volume and rate messages exchanged can become much more important than with the current BGP. Several distinct routing and forwarding tables will have to be maintained per router. This number will depend on the number of *Meta-QoS-Classes* supported.

ASBR routers will have to swap DSCP according to binding rules of established pSLS.

Shaping and policing will probably impact the router forwarding performances.

Deployment of the QoS Internets can be gradual and assumes a close cooperation of adjacent providers.

7.3.1.9 Deployment issues

A new qBGP has to be specified, developed and validated.

IGPs will compute routes based on the destination prefix information AND *Meta-QoS-Class* (l-QC for *Light approach*) (probably QoS routing planes identified by a couple `{QC, MC}`).

Routers will have to be updated.

Introduction of such a new service might be risky and slow.

7.3.1.10 Requirements on pSLSs

Within this solution option, a pSLS should be considered as a permission to send some maximum agreed volume of traffic, towards any destination, within the context of a given *Meta-QoS-Class*.

Before establishing any pSLS, an ISP shall qualify its l-QC in verifying their compliance with *Meta-QoS-Classes*. Only l-QCs for which a *Meta-QoS-Class* membership has been stated, are eligible to be extended across the Internet. This is necessary to ensure the service extension concept.

pSLS will likely be negotiated with some contractual maximum bandwidth per *Meta-QoS-Class* (l-QC binding in the case of the *Light approach*). Consequently, the upstream AS should make sure it doesn't send more data than it is allowed to. The downstream AS must police the incoming traffic so that it fits in the contracted traffic envelope.

The routers automatically choose the path. pSLS invocation and contractual bandwidth consumption will be hard to achieve.

7.3.1.11 Implications for cSLSs

Within this solution option, a cSLS should be considered as permission to send some maximum agreed quantity of traffic, towards any destination, within the context of a given *Meta-QoS-Class*.

Network accessibility through a *Meta-QoS-Class* plane is never permanently ensured.

The implicit versatility of QoS value shall be indicated. Informational values can be provided by the reporting functionality added to qBGP. These values can't be contractual.

cSLSs don't need to explicitly state in advance the destination points.

The result is a best-effort QoS service. Normally clients should get the level of quality they need. But, we can't guarantee there will be no disruption or big fluctuation in the QoS they receive.

7.3.1.12 On demand inter-domain pSLS interactions

As described above, this approach allows a set of MC routing planes to be built dynamically; QoS information is exchanged within each plane for route computation purposes, with the final objective of selecting optimal QoS paths that meet average customer application needs.

Thus, if a remote domain does not support an appropriate pSLS that extends a given *Meta-QoS-Class*, it may imply, from a local domain perspective, the introduction of possible holes in the address space within the corresponding *Meta-QoS-Class* plane.

In order to solve this issue, one of the potential solutions is to make use of an “On Demand” pSLS feature to request the establishment of the missing *Meta-QoS-Class* extension class near the domains where these "QoS holes" exist.

The reasons why a remote domain may have no pSLS established for extending a Meta-QoS-Class plane are mainly of 2 categories:

- The remote domain cannot do it: because no DiffServ architecture has been deployed in its domain or extended MESCAL protocols and mechanisms are not available in its domain. Nothing can be done in that case. This domain can only be reached or crossed on a best effort basis.
- The remote domain doesn't want to do it because he hasn't identified yet any valid business reason for doing it.

In this latter case, it is suggested that the provider anyway proceeds to a QC-mapping and QC-binding operation and activates, at its domain's boundaries, specific qBGP functions (to be specified) allowing to advertise lifeless o-QC (lo-QC) he would be ready to implement (QC-implementation) if some interest was shown by external providers In turns, these lo-QCs could be used by external domains and propagated using qBGP. From a single domain standpoint, qBGP could announce:

- Either a lifeless QoS reachability for a given destination within a Meta-QoS-Class plane with the corresponding lo-QC
- Or an e-QC and a possible lo-QC when this lo-QC would have been selected by qBGP if this lo-QC hadn't been a virtual one.

Thus, thanks to this mechanism, external domains can become aware of possible capabilities of a remote domain and can now identify this domain quickly so that an On Demand pSLS can be requested easily and a negotiation cycle started.

7.3.1.13 Applicability to the Business Model

The business target covered by the basic approach is clearly the residential market. It is suitable for service providers who are willing to benefit from network-wide differentiated services for improving their existing services or as a leverage to create new ones. This can be the case of web-based services (e-learning, e-training, consultation services…) or video-on-demand for instance for which some categories of end-users are ready to pay to get better services. The approach does not constrain customers to specify the final destination of the traffic in the cSLS (or pSLS between providers). The address space, which can be reached within a Meta-QoS-Class (or l-QC) plane, depends on the number of established pSLSs between providers. All Internet users would consequently not be able to request such services until it is globally deployed.

The basic approach is resilient, scalable and respects the underlying philosophy, which guided the elaboration of the Internet. But the QoS guarantees it provides are loose since:

- QoS performance associated with an e-QC can change at any time since the Inter-domain path can change.

It is impossible to provide end-to-end bandwidth guarantees. The traffic matrix can be very stochastic (destination addresses and routes followed) and network engineering can only be achieved on a statistical basis.

7.3.2 Statistical Guarantees Solution Option

This section presents how to build the required capabilities in order to be able to support end-to-end QoS classes (QCs), and it focuses on the required inter-AS interactions. These classes can be used to offer end-to-end services with some statistical guarantees.

7.3.2.1 Introduction

Each domain is engineered to support some Quality of Service classes, also known as Per Domain Behaviours (PDBs) [Nichols01].

The engineering of QoS classes includes the provisioning of network resources in terms of routing and bandwidth management (including scheduling and buffer resources) for implementing the required Per-Hop-Behaviours (PHBs). This provisioning can be done either by an automaton (e.g. [Trimin01]) which defines the appropriate provisioning directives and enforces them to the network elements, or through human static configuration. Even in the latter case there may be tools, which aid the human administrators to take the provisioning decisions (e.g. [Feldm00]). We have to mention that in this engineering for provisioning process, we include the over-provisioning engineering model. In this solution option the desired behaviour of some class is based on allocating link bandwidth, which is well above the maximum average requirements for that class (common practice is to keep it the utilisation below 50%). In the latter engineering model still some basic differentiation between classes is assumed to exist, but the over-provisioning factor between the classes may vary according to the significance of the class (e.g. a premium class may be over-provisioned to always below 10% utilisation).

Note that this solution option does not take into account the access network QoS capabilities in the forwarding path. These capabilities can be incorporated into this solution option either if the first hop ISP takes into account the QoS capabilities of the customer's access network, or the access network itself plays the role of an AS, as this role is defined by this solution option.

The timescales in which these engineered classes are realised and possibly changed, are at the level of a Resource Provisioning Cycle [Trimin03], which is from few hours to the level of weeks, depending on the operating procedures of the providers. This is the medium-to-long timescale traffic engineering as defined by the IETF [Awduc02]. Normally these classes are not expected to change considerably from one provisioning cycle to another because the provider will have agreements based on these classes which impose some restrictions on the supported classes. A provider will always try to enforce these classes by setting them as the engineering target QoS classes (see below for more details).

QoS classes are differentiated within an AS by using a different DSCP (Differentiated Services Code Point) value in the appropriate octet (Type Of Service TOS → IPv4, Traffic Class → IPv6) of the IP header. This DSCP marking is then used to classify the packets into (ordered) traffic aggregates which are processed (buffered and forwarded, typically) according to different PHBs, depending on the class.

The solution option described in this section makes use of a concept the Virtual QC, in addition to the QC concepts presented in section 4.2.2.

Virtual QC (v-QC): this is a virtually introduced engineered QoS class. Within an AS, the differentiation of packets into l-QCs is implemented using a different DSCP for each l-QC, which then maps onto the one PHB. This means there exists a "1-1" mapping between a DSCP an l-QC and a PHB. If we relax this "1-1" mapping, and allow for "N-1" mappings, i.e. *n* DSCPs mapped to the same

PHB, it would be as if *n*-1 additional l-QCs were introduced. We call these additional l-QCs, virtual QCs (v-QCs). Note that the mechanism to support v-QCs already exists since the DiffServ standard supports this "N-1" mapping from DSCPs to PHBs. The need for introducing these v-QCs, the rules for their introduction, and their use in this solution option is going be discussed in the following sections (see section 7.3.2.4). Because a v-QC is at the same level as a l-QC, in the rest of this document we may use the term l-QC for both of them and will differentiate only when necessary.

7.3.2.2 The Cascaded Solution for Statistical Guarantees

The essence of the MESCAL solution to service option 2, i.e. offering services with some statistical guarantees, can be summarised as follows:

- The end-to-end QCs are built based on the cascaded model, i.e. by service peering between adjacent ONLY domains.
- It supports statistical end-to-end guarantees both in terms of QoS parameters and in terms of bandwidth.
- The solution requires the pSLS to valid for specific address prefixes. A pSLSs includes the required QoS class, bandwidth both with some probabilistic guarantee, for some specific destinations. The service peer AS that signs to a pSLS, undertakes the responsibility to adhere to all the agreed requirements, within the error margin given by the probabilistic guarantees terms.
- An AS that wishes to offer a particular o-QC to a destination prefix, is allowed to use MORE THAN ONE e-QCs, i.e. many internal l-QCs and many external o-QCs, as long as the offered o-QC constraints are met.
- Mapping and binding are allowed on an N-M basis. This means that, in order to build a given o-QC which satisfies business objectives, the solution option allows the mapping process to produce a set of e-QCs formed with different l-QCs and different external o-QCs. Constraints on both can be imposed by the business objectives. These objectives MAY (not necessarily though) facilitate the Meta-QoS-Class concept.
- The total number of offered o-QCs, both e-QCs and l-QCs, is constraint to be NO MORE THAN 64, since the Differentiated Service Code Point (DSCP) is the means to indicate both internally and externally one QC in an IPv4 realm. This controls the scalability of the solution unless IPv6 was introduced in the network.
- The QC splitting problem is tackled with the v-QC, an engineering approach that facilitates the fact that we can configure the routers so that several different DSCP markings refer to the same Per-Hop Behaviour.
- Inter-domain routing is pSLS constrained, i.e. the established pSLSs influence the routing. Inter-domain routing is also QoS-enabled, i.e. it is able to compute different paths for different QCs. There is no other mandatory requirement from routing.

7.3.2.3 QC Advertisement

QC advertisement is not mandatory for this solution option, but QoS-related information needs to be propagated throughout the (peering) domains. This means that this solution option can use the advertised o-QC information of adjacent AS but it is not a requirement to have such advertisements that is we can have a fully operational solution even without advertisements.

In the following sections it will be clear that QC advertisements are only necessary when we want to make a mapping between the QCs so as to find the ones that are compatible and then based on some logic request a pSLS. In the case we do not have explicit advertisements, the logic which decides how to request a pSLS will base the decision only on the requirements set locally and then the pSLS receiver will do a "best match" counter proposal during the pSLS negotiations.

7.3.2.4 QC mapping

We have QC mappings at two levels. The first level is mapping within the AS, between the local l-QCs or e-QCs and the o-QC. The second mapping is an external mapping between the l-QCs and/or e-QCs of one AS with o-QCs of the adjacent ASs. In this solution option in the case where we do not have any QC advertisement or discovery the QC mapping step is omitted.

Both mappings are dictated by the fact that an AS wants to extend its own local l-QCs to prefixes that can be reached by traversing other ASs. In the example shown in Figure 33, AS1 wants to extend the offering of QCs to addresses located outside of the domain, in this example to addresses located in AS2. Somehow AS1 needs to communicate (see section on pSLSs) the requirements it has in terms QoS. The mappings described in this section can be either the result of an agreement between the two adjacent AS, in the case of AS1 does not know the o-QCs of AS2, or could be done before any agreement based on the information that AS1 has about the o-QCs of AS2. In this example we are showing the mapping being done with the o-QCs of AS2. These o-QCs maybe composed of many e-QCs or l-QCs, in both cases there will be an l-QC applied internally to AS 2. The latter are the l-QCs shown in the figure and thus they represent either a single used l-QC or the first part of an e-QC.

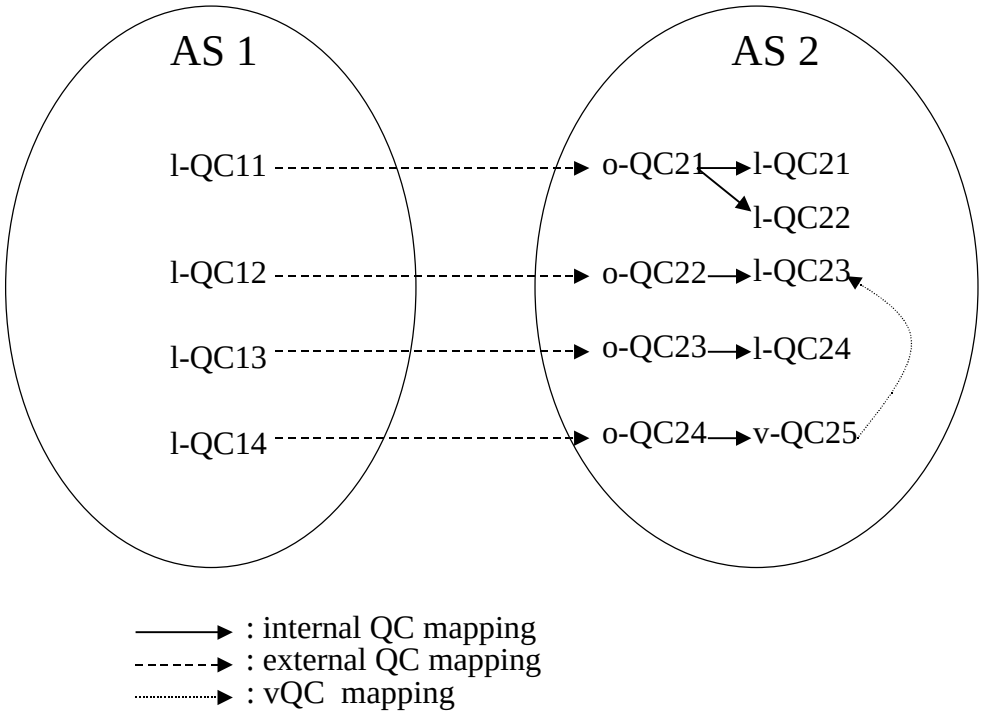


Figure 33: QC Mapping example

The internal mapping between the o-QCs and the QCs of a given AS is a "1-N" mapping. This means that the ISP has the freedom to provision any number (less than 64 in an IPv4 scenario, since they have to be uniquely identified by a DSCP) of l-QCs but only offer some of them to the peer ASs. For example o-QC21 in AS2 (see Figure 33) is mapped to both l-QC21 and l-QC22. The rules for such mapping is that all the l-QCs mapped to the same o-QC must be "compatible" with each other and the o-QC, and in addition the following must be hold:

$$\forall QC_i \rightarrow o\text{-}QC, \quad o\text{-}QC \geq QC_i \tag{1}$$

i.e. each l-QC used to support an o-QC it must be *at least as good as* the o-QC it is mapped to. The reason for having additional l-QCs used by same o-QC is for reasons like load sharing or offering a much better QC to internal customer VPNs. It is not compulsory for every l-QC to be mapped to an o-QC. When more than one l-QC is used, then there must be some static or dynamic load balancing of the o-QC traffic to the various supporting l-QCs. Note that in the example shown in the figure we do not show the internal mapping of l-QCs to o-QCs within the AS1.

The AS, e.g. AS1, that requires the extension of its own l-QCs to the addresses supported by the peer AS, e.g. AS2, may request to map an o-QC for which the receiving AS2 does not have any advertise an offered QC. For example l-QC14 does not have a compatible l-QC within AS2. In this case AS2 may refuse this mapping and this will create a "hole" in the end-to-end QoS, and therefore AS1 will only be able to support this class for its own addresses. On the other hand, AS2 may want to offer a mapping to AS1 for one of AS2 l-QCs, e.g. l-QC23, which is *at least as good as* the l-QC14.

In the latter case, we may have the splitting problem, see section 5.5.1. This problem will only occur when the traffic is going to exit AS2 towards another AS, and in this case AS2 will not be in a position to know which part of the l-QC23 aggregate was from l-QC12 and which from l-QC14. One solution to the splitting problem is to allow only merging of l-QCs and never splitting. In many cases this solution may not be acceptable, since the end-to-end classes, which were merged at some point, will tend to be the same as the path includes more ASs.

We propose a more general solution to the splitting problem by introducing the notion of a virtual QC (v-QC). For example when AS2 receives a request for mapping the l-QC14 from AS1, and realises that the closest local QC support this request is l-QC23, for which there already exists a mapping to another offered QC, i.e. o-QC22, it will introduce a new o-QC, o-QC24, which is identified by a unique DSCP. The role of this new DSCP is only to differentiate between the two o-QCs since the corresponding PHB received by the packets of both classes will be the same, i.e. that of QC23, but the two classes will be distinct at every egress point of AS2.

The above describes unidirectional mappings from AS1 to AS2. Similarly, AS2 will request information from its peer AS1 to extend its own l-QCs to the addresses supported by AS1, thus providing unidirectional mappings from AS2 to AS1. The approach for mapping will be similar to the one described above. At the end of the day we will have mappings for both directions.

7.3.2.4.1 Mapping with Meta-QoS-Classes

An observant reader may have noticed that with the mapping procedures discussed so far, if everybody accepts their peers requests for all o-QCs mappings by introducing v-QCs, we will end up with a large number of required l-QCs within each AS. At the steady state of the overall mapping procedure, the number of o-QCs within each and every AS, will be the same to that of the AS which has the maximum requirements in o-QCs. This number has to be bounded by 64, since the DSCP value must uniquely identify each o-QC within an AS, it may still be big enough to introduce complexity and scalability concerns in negotiations, provisioning, and routing functions.

In order to further reduce the total number of o-QCs and at the same time adhere to some globally well-known and defined classes, we make use the notion of the Meta-QoS-Class, see section 7.3.1.1.

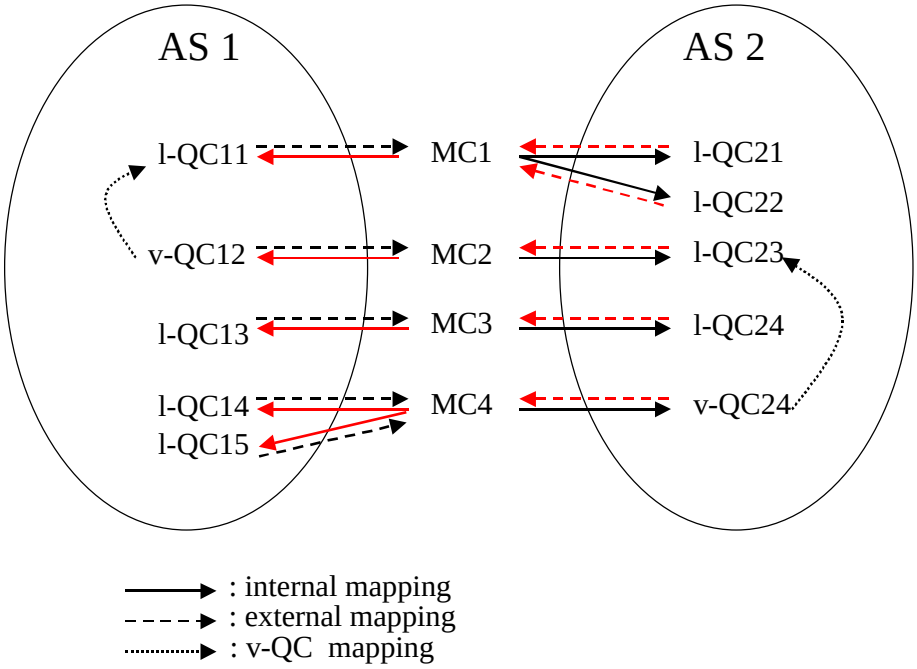


Figure 34: Mapping example with Meta-QoS-Classes

In Figure 34 we show the mapping example presented in the previous section with the use of MCs. The figure shows the mappings from both sides, with blue corresponding to the mapping agreement for traffic from AS1 to AS2, and the red for the mapping agreement in the opposite direction. Note that in this example since we are showing the mappings in both directions we can observe the QCs within AS1, which constituted the QCs of the example shown in Figure 33. We can observe that, as with the l-QC21 and l-QC22, the l-QC14 and l-QC15 are mapped internally to the same MC. The actual mapping of incoming traffic to l-QC14 or l-QC15 can either be done statically or dynamically with load balancing between the different l-QCs, which can be relied on load sharing criteria and implemented at the with a hashing function criteria.

7.3.2.5 QC binding

QC binding is the application of the bind operator "$\oplus$" between QCs, in order to define an e-QC. The ultimate target is to have at each ASi a precise definition of the e-QCs that are available. That e-QC can then be offered, i.e. become an o-QC, to the other upstream ASs. In general an o-QC can be either an l-QC or an e-QC.

The binding between the QCs is done in a *cascaded* fashion. This binding is the recursive definition of e-QCs at ASi, as follows:

$$e\text{-}QC^0 = l\text{-}QC^0 \tag{2}$$

$$e\text{-}QC^i = l\text{-}QC^i \oplus o\text{-}QC^{i-1} \tag{3}$$

That is an $e\text{-}QC^0$ at the home AS of the address prefix is defined to be a local $l\text{-}QC^0$ of that AS. And then recursively define the $e\text{-}QC^i$ of an ASi is the binding result of a local $l\text{-}QC^i$ of that AS, and an offered $o\text{-}QC^{i-1}$ of the previous ASi-1. This cascaded definition of QCs is the main characteristic this solution option.

According to the definition for e-QC as given above, if we bind different l-QCs internally with the same external o-QC then the resulting e-QCs will be different, similarly if we bind the same l-QC internally with the different external o-QCs the resulting e-QCs will be different. This solution option

does not restrict these bindings, and they are all allowed, thus it allows N-M bindings. Restrictions can only apply based on the policies of the domain.

When based on marketing and business objectives, the service planning functionalities an AS*i* decides to offer an o-QC towards some destination, this o-QC will have specific characteristics. It may be the case that more than one e-QC are able to comply with the requirements of the specified o-QC. So the $o\text{-}QC^i$ can utilise all the $e\text{-}QC_k^i$ which are at least as good as it:

$$o\text{-}QC^i \rightarrow e\text{-}QC_j^i \tag{4}$$

such that

$$e\text{-}QC_j^i \leq o\text{-}QC^i, \quad \forall j \tag{5}$$

The actual offering of the o-QC happens when this is included in pSLS, i.e. when AS*i* becomes the downstream AS for an AS*i*+1. In this case AS*i* has to make some selection about which bindings in effect, that is to choose which of the compliant $e\text{-}QC_j^i$ will be used for offering that o-QC. In the simplest case a single $e\text{-}QC_k^i$ can be the chosen one. In a more complex scenario there may be some policy to have more than one in effect, so as to allow for some dynamic load balancing between the locally used l-QCs and the agreed with the downstream o-QCs, as those are bound in the definition of each of the e-QCs.

If the exists such a load sharing functionality as discussed above it will have to take into account the utilisation of the various $l\text{-}QC_j^i$ s and $o\text{-}QC_j^{i-1}$ s bound as in (3) to the $e\text{-}QC_j^i$ s which belong to the subset of the e-QCs which are compliant to $o\text{-}QC^i$. The implementation of the load balancing decision, i.e. splitting ratios and mapping of traffic could be done in two ways. Either at the forwarding level based on some hashing function on the fields of the IP header, or at a higher level based on assignment of SLSs to each of $e\text{-}QC_j^i$ s bindings. In any case this load balancing could be considered in combination with the load sharing options discussed in section 5.4.2.

Summarising, the QC binding operation includes the following sub operations:

- Out of all possible QC mappings we need to select the ones for which will establish pSLSs
- When we are to offer an o-QC and accept a pSLS from an upstream AS, the pSLSs with our downstream ASs must be in place. At this point we need to decide out of these pSLSs which are the ones that will be used for offering the particular e-QC.

And finally, the actual QC values used with binding operator "$\oplus$" are the ones decided by any dynamic load balancing algorithm.

7.3.2.6 QC Implementation

The basic assumption of this solution option was that, within an AS, the packets belonging to a QC are uniquely identified by the DSCP value marking in the IPv4 TOS, or IPv6 Traffic Class fields. Packets of the same QC have the same DSCP marking.

Between ASs, this solution option proposes to use the same field, i.e. the DSCP, to signal the o-QC mapping. The exact DSCP values that will be used to signal the o-QC requirements between the ASs are defined between the ASs during the agreement request-negotiation process.

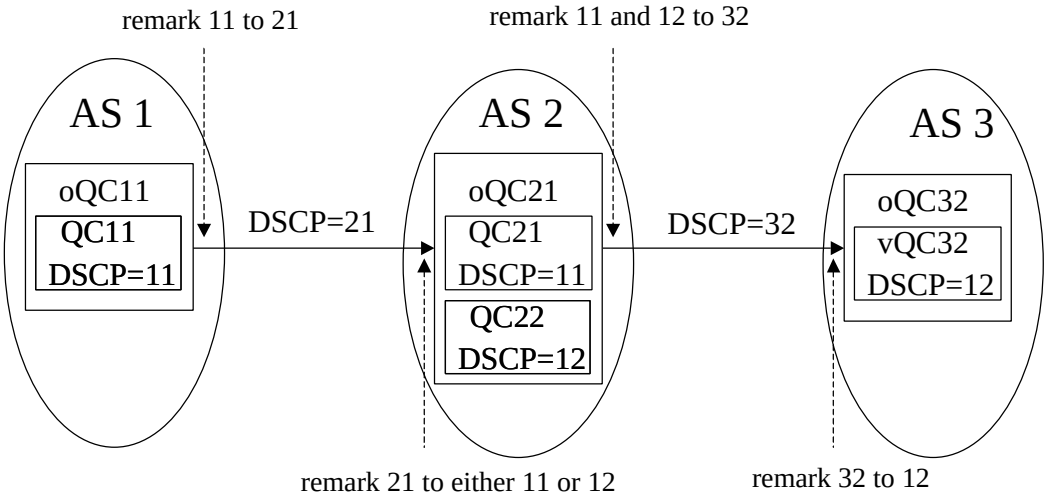


Figure 35: QC implementation example

In Figure 35 we show a QC implementation example between 3 ASs. AS1 internally maps the o-QC11 to QC11, which is identified by the DCSP value 11, AS2 maps internally QC21 and QC22 to o-QC21 and uses DSCP values 11 and 12 respectively. The external mapping between o-QC11 and o-QC21 is signalled with DSCP value 21, that is when traffic marked with 11 leaves AS1 it is remarked to 21, and when traffic marked with 21 enters AS2 it is remarked, either statically or dynamically (for load balancing) to 11 or 12. Similarly, the traffic which leaves AS2 and is marked 11 or 12 is remarked to 32 in order to obey the external mapping of o-QC21 to o-QC32.

7.3.2.7 Requirements on pSLSs

Thus far we have assumed that one AS is able to use a peer AS's offered QC. This ability is defined in a pSLS (peer-SLS). The purpose of this section is to identify at a high level the required fields in a pSLS for the solution option in consideration. The details of the pSLS structure will be subject of further research within the MESCAL project.

This solution option requires building QoS agreements only with direct peering ASs. Thus the pSLSs will be requests for agreements with the management/administrative entities of the peer ASs. The number of offered o-QCs that an AS is supporting is defined by the Marketing and Business objectives of the ISP, and, in this solution option, is constrained by the total number of DSCPs, i.e. 64.

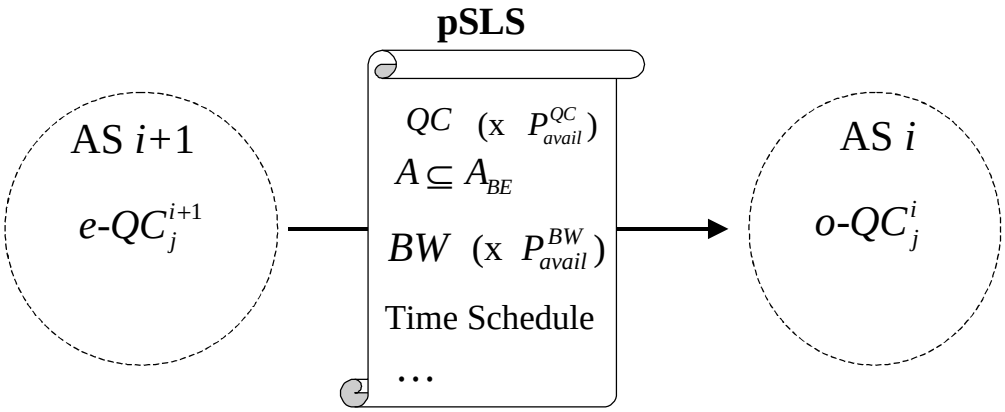


Figure 36 Abstract required fields in a pSLS

With reference to Figure 36, we assume that AS i+1 wants to create an $e\text{-}QC_j^{i+1}$, by extending its $l\text{-}QC_j^{i+1}$, that spans to addresses other than the ones managed by itself. We distinguish between two cases:

1. There have been QC advertisements, and thus AS *i*+1 knows the o-QCs offered by AS *i*.
2. There have not been any QC advertisements, or there were "marketing-language" advertisements, e.g. "I offer a low delay QC", without specific values in the advertised QCs.

In both cases the decision for creating the $e\text{-}QC_j^{i+1}$ is driven by the Business/Marketing policies of AS *i*+1. The difference is that in the first case the actual *QC* value that will be *requested* in the pSLS is decided by choosing the most appropriate from the ones that have been offered and advertised by the adjacent AS, while in the second case, the request is arbitrary and the requested AS will find the one of its offered QCs which is the closest to the request.

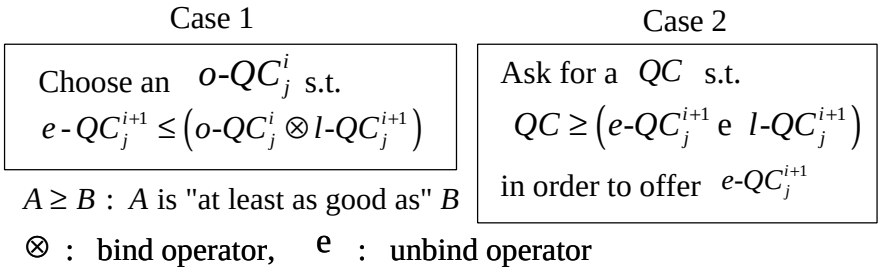


Figure 37: Two cases for requesting the QC in a pSLS

In Figure 37 we show in pseudo-code the difference between the two cases for requesting a pSLS mentioned above. In the first case the requester chooses to ask for the o-QC of AS *I* which bounds with the local QC that can produce an e-QC which is at least as good as the required e-QC. In the second case, it is free to request whatever it wants, which is the QC which is at least as good to the target e-QC unbound to the local QC. Note that in both cases after negotiations and based on the pSLS receiver AS's policies, the end result in the pSLS agreement is one of the offered QCs, e.g. $o\text{-}QC_j^i$, of AS *i*.

AS *i*+1 knows the list of address prefixes A_{BE} that it can reach via AS *i*, e.g. from the routing protocol information used to compute Best Effort (BE) routes, the pSLS should include a list of addresses prefixes $A \subseteq A_{BE}$, to which the requested e-QC should be available. AS *i* will know the answer to this question based on its own addresses and on the pSLS that has been established with its peers.

The bandwidth is another important factor that should exist in a pSLS. This request will be based on the predicted requirements of AS *i*+1 for that peering connection. This information will help AS *i* to provision its own domain. It is expected that bandwidth is going to be renegotiated more often than the other fields of the pSLS. The time schedule of the required QoS is also another important parameter in the pSLS.

We envisage that there will be a number of negotiations between the peers before an agreement is reached. The end agreement will have to include the DSCP value that will signal the class mapping. In addition the whole agreement, and/or individual fields, e.g. e-QC, BW, may have an availability factor.

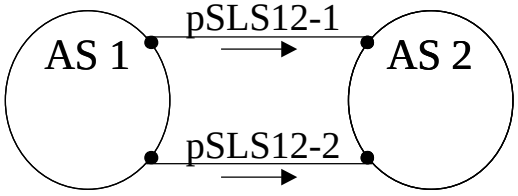


Figure 38: Peering at more than one point

If we have more than one peering points between two ASs we require to have a different pSLS for each of them. Even if we want to use the same o-QC from both peering points we need to have different pSLSs.

Note such a pSLS is the basic building block of agreements between two ASs. This means that this may be only part of the final agreement that needs to be negotiated, and may include more than one of the basic building block pSLSs. Thus the amount of required communication and negotiation can be significantly reduced, since we they will be done on higher level than the basic pSLS. For example in Figure 38 AS1 may choose to put pSLS12-1 and pSLS12-2 into the same agreement and request for a combination, and in this case AS2 will be in position to make alternative suggestions, especially as far as the total bandwidth is concerned.

Summarising, the following are the required fields in a basic pSLS:

- A QC, which corresponds to a peering offered QC, e.g. oQC_j^i (*Which quality?*)
    - An associated probability with which the QC is available, P_{avail}^{QC}
    - The DSCP which signals the agreed QC
- The address prefixes which are covered by the pSLS, $A \subseteq A_{BE}$ (*Where to?*)
- The bandwidth available for use BW (*Which quantity?*)
    - An associated probability with which this bandwidth is available, P_{avail}^{BW}
- The time schedule

7.3.2.8 Scalability

This solution option follows the cascaded model for building end-to-end QoS and constrains the maximum number of offered QCs to be no more than the maximum allowable DSCPs. In the following we will make a first attempt to estimate scalability imposed by the solution option in the QC management and routing decision process, as well as the routing dissemination process.

When this solution option is used in an IPv4 or IPv6 realm, it does not constrain the possible combinations between the QCs. But it allows in an IPv4 realm only 64 QCs to be offered per AS. This means that in the worst case the possible combinations for offering a single QC is 64xN where N is the number of peer ASs, and thus 64^2xN for offering all the possible QCs. This number will has to be multiplied for each AS pair with number, K, of peering points between that pair of ASs. So, the scalability factor for supporting and offering the maximum number of QCs is in the worst case 64^2xNxK, assuming that K is the maximum number of different peering points between two ASs.

The number given above is the scalability factor for the inter-intra domain routing decision processes (including load balancing) as well as the QC management ones. This means that in the worst case the routing information that is handled today will have to be multiplied by scalability factor 64^2xNxK. This is the worst case that assumes that all peer ASs offer the maximum (64) number of QCs and that we have pSLSs with all our peers ASs in order to use all their offered QCs. Also it assumes that all ASs peer with the AS in question at the maximum number of points, K. For the routing information dissemination process (section) the scaling factor is 64.

We can see that the solution option, apart from the constant factor, scales with the number of peer ASs and the number of peering points for each peer AS. Since the larger, e.g. tier-1, tier-2, the ISPs the more the peering points and thus smaller ISPs can handle the burden for supporting the QoS. We can also observe that the scale factors are only first-degree polynomial to the number of peering points and peers ASs, thus avoiding an exponential growth.

Note that this scalability assessment is at an abstract level, and does not include any considerations about the IPv6 realm case. After the exact algorithms have defined will be in position to make a more detailed assessment of the scalability factors.

7.3.3 Hard Guarantees Solution Option

The level of QoS guarantees reached with the above options is not satisfactory for all corporate business applications or services for which strong guarantees must be provided.

In particular, such categories of end-users would request:

- Guaranteed QoS performance
- Bandwidth reservation

In order to satisfy these requirements it is necessary to elaborate a solution, which enhances the MESCAL solution for loose and statistical guarantees service options, in two ways.

- The first action is to fix the inter-domain path so that the QoS performance of an e-QC cannot change.
- The second action is to provision the requested bandwidth all along the path followed by the datagram to reach the final destination. This reservation must be achieved, in a coordinated manner, within all crossed domains and at the boundaries of these domains.

Those two constraints imply that the final destination of the traffic is (a priori) known to the provider and will become mandatory information for all cSLS established within this context.

MPLS is a technical solution to forwarding and bandwidth reservation, and has been successfully deployed by a large number of providers for supporting connection oriented services such as IP VPN services for which traffic isolation criterion was the highest need. Then, the solution evolved to encompass QoS issues, and Traffic engineering functions were progressively introduced. Up to now, some providers have deployed MPLS TE but only within their own domain.

Nowadays, services are deployed on a same basic infrastructure (best-effort shared IP network) on which more elaborate functionalities (MPLS for instance) rely for providing enhanced network services mainly intended for specific corporate customers or providers needs. These extra functionalities were introduced because the basic IP approach failed to support those added-value services or was not considered to be efficient enough.

Within the previous sections we defined a "QoS enabled shared IP network". Thus, in the same way additional functionalities were built on top of best effort network, we propose to extend the basic QoS approach we defined in section 7.3.1 with additional functionalities. Inter-domain MPLS TE is a good candidate since it entails most of the features we need and it has strong business support.

7.3.3.1 Overview

Standardisation work is currently going on to extend the MPLS TE approach to the inter-domain case. This is in particular the case of the editing effort lead by Cisco who submitted in the IETF a set of Internet drafts on this subject [Vasse03].

Summarized in few words, this proposal will allow a provider to establish end-to-end Label Switched Paths (LSPs) across multiple ASs. The solution is currently elaborated for solving two important future customer requirements:

- Traffic isolation (for transparently crossing several ASs: inter-as transit traffic, virtual lease lines, IP VPN…)
- Bandwidth protection

In the approach described in [Vasse03], two main modes for establishing LSPs have been defined:

- A static loose hop approach: in which an inter-domain path is defined as an ordered exhaustive list of all the Label Switched Routers (LSRs) in the path towards a destination or a subset of them containing at least the AS Border Routers (ASBRs). This list is then used in an RSVP-TE LSP path set-up message for establishing the LSP.

- A dynamic mode: in which each LSR receiving an RSVP-TE LSP path set up message will have to determine automatically the next hop ASBR, based one the IGP/BGP reachability of the TE LSP destination.

If for some reasons (technical or administrative) the LSP cannot be established, an RSVP error message is returned and another path must be computed.

In these two modes, path computation is a real issue, even considering the deployment of this solution over a non-QoS enable IP network. The dynamic mode doesn't describe how the LSR will compute the path and the static loose hop approach reports all the difficulties of the path computation on set of network administrator or on an external entity called the Path Computation Server (PCS).

MESCAL introduces QoS consideration in this MPLS TE approach and embeds its IP based approach in such a way the QoS MPLS TE solution can greatly benefit from the underlying infrastructure to make easier the computation and the establishment of QoS LSPs.

7.3.3.2 QoS and LSP considerations

In a best-effort environment, the establishment of a best-effort LSP between two extremities (identified by their respective IP address) is only constrained by the existence of an inter-domain path (learned via IGP/BGP) providing that network resources are available.

Within an intra-domain QoS context, each LSR is configured to support the PHB corresponding to the l-QCs defined by the provider for its domain. If we ignore resource availability observations (bandwidth for instance) each datagram conveyed within an LSP will be handled according to the l-QC it requests, whatever the path it follows. In particular, this means that a given LSP could be multi-coloured, that is it could be used to transport datagrams requesting different l-QC from one point to another.

In the inter-domain QoS context, it is not possible anymore to assume that a single LSP will cross a set a domain in which compatible l-QC will have been defined. All l-QC defined by the provider requesting the LSP may possibly have been extended up to the target destination termination of the LSP but the path to follow can be different.

Some provider's customer will likely ask for multi-coloured LSPs (or EXP-Inferred-PSC LSPs as defined in RFC3270). Sometimes it will be possible to aggregate all requested l-QC traffic in the same LSP, sometimes not. Several LSPs will have to be used. Thus, a service offering will have to take into account a possible multiplication of the number of LSPs to establish, in order to satisfy customers' requirements. The solution will consequently have to support the ability to compute multi-coloured path for an LSP with some options allowing returning either one multi-coloured path or several mono/multi-coloured paths as the result of a QoS path computation.

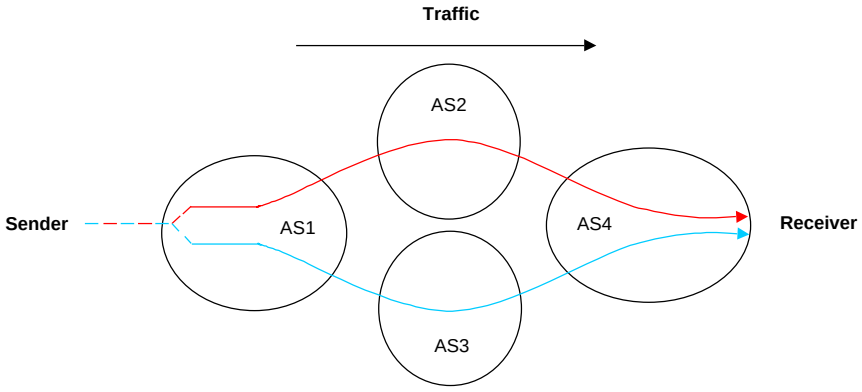


Figure 39: Multi mono-coloured LSP

At the service access point, in a situation where several mono-coloured would have been created, (first LSR), the injection of the traffic in the correct LSP would have to take into account the destination address of the datagram **and** the requested l-QC.

In the above figure, traffic splitting occurs at the boundary of the first domain but it could be imagined that this splitting is achieved at an inter-domain boundary, but this case is for further studies.

7.3.3.3 Working overview

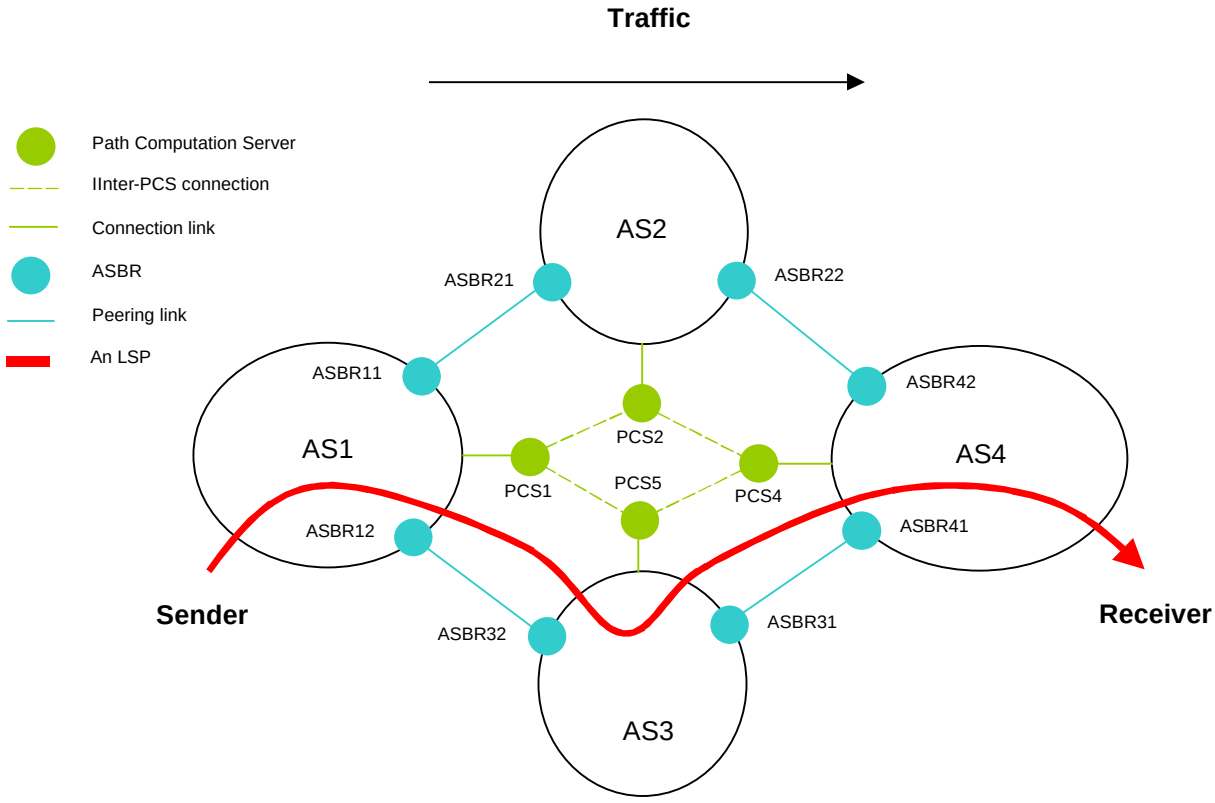


Figure 40: Working overview

Each domain is assumed to have some l-QC deployed. qBGP is running between domains which have agreed to established a pSLS. Each domain receives, per Meta-QoS-Class plane, the set of destinations that can be reached within each *Meta-QoS-Class* plane it supports, together with some aggregated QoS performance information. A full iqBGP (Internal qBGP) mesh between all ASBRs of a domain has been set-up so that destination learnt by a peering ASBR can be propagated to the other ASBR of the same domain. QoS routes learned by qBGP are made known of the qIGP in place in each domain so that a datagram can be routed up to the correct egress point within a *Meta-QoS-Class* plane.

A PCS is present in each domain and receives qBGP announcements from all ASBR of the domain. Thus, the PCS can know all destinations that can be reached within a *Meta-QoS-Class* plane together with their associated QoS performance characteristics. Moreover, each PCS establishes a session with the neighbours PCS of the external peering domains for which pSLS have been contracted (the protocol between PCS needs to be specified but is temporarily named: Path Computation Protocol: PCP). Communications between PCS occur within the best-effort Meta-QoS-Class plane.

For creating an inter-domain QoS LSP, the domain which requests the establishment of the LSP asks its PCS to compute an inter-domain path satisfying QoS constraints expressed in term of *Meta-QoS-Class* availability along the path and optionally an associated bandwidth guarantee per *Meta-QoS-Class*. This first PCS selects one possible path among the set of possible alternatives and identifies the next-hop domain. It then verifies that appropriate resources are available in its own domain and set-up administrative pre-reservation in the management system of the domain. Then it contacts the next hop PCS in the external domain, requesting a path computation between its peering ASBR and the termination address of the inter-domain LSP. This second PCS performs the same computation as the first one did and the procedure is iteratively repeated up to the last PCS. If a path satisfying all

requirements is found, each PCS returns the QoS path to follow as a list of LSR. Each intra-domain sub-path is concatenated with the result received and when the last result reaches the originating PCS the whole path is available. A PCS can try several alternatives before sending back any path error computation. If all PCS return an error the LSP cannot be established. Otherwise, an RSVP TE LSP paths set up message is sent by the originating LSP termination, with a whole computed path or with some loose hops, if some of the sub-paths returned were incomplete.

When the RSVP TE Resv message is returned, some outsourcing admission control should be done at each inter-domain boundary in conjunction with information stored by PCS in the management system, for security, provisioning and accounting purpose.

7.3.3.4 QoS path computation

As briefly described above, path computation is distributed between a set of cascaded PCS. At a high and preliminary description level PCS communicate together thanks to the following three basic messages (connection, identification and other security issues are not considered here):

- Path Computation Query Message (PCQM): sent by the requesting PCS
- Path Computation Result Message (PCRM): returned by the queried PCS. The result may contain a complete path or a loose path (path containing a non exhaustive list of LSR hops)
- Path Computation Error Message (PCEM): when an error occurs or a path cannot be found.

A PCQM is issued by a PCC (Path Computation Client) embedded in a LSR, a PCS or any other relevant entity.

The PCQM contains the identification of the terminations (starting point (LSR or ASBR) and ending point (LSR or ASBR)) of the LSP. Together with those connection-oriented parameters, a set of additional QoS information can optionally be provided:

- Identifiers of the Inter-domain *Meta-QoS-Classes* requested
- Associated bandwidth guarantees requested
- Multi-coloured LSP or not.

When queried, each PCS performed the following operations:

1. Find an egress point. If it cannot the PCS returns a PCEM.
2. Compute an intra-domain path to reach the targeted egress point.
3. Check if appropriate resources are available in intra-domain and at the inter-connection link level. If it cannot procedure restarts in 1.
4. Administratively reserves the corresponding intra-domain resources. If it cannot the procedure restarts in 1.
5. Contact the neighbour PCS which computes the remainder of the path. If it cannot administrative pre-reservation are freed then procedure restarts in 1 or can return a PCEM depending on the domain policy or additional parameters of the initial query.
6. If the returned result is an error, administrative pre-reservation are freed and the procedure restarts in 1. If the result is a path, a resulting path is formed in concatenating the external path with the local computed path. Then the PCS returns its result.

7.3.3.4.1 Finding an egress point

Since it receives qBGP information, each PCS, as any ASBR of its domain, knows all the available QoS routes for reaching a particular destination within a *Meta-QoS-Class* plane. The LSP is not constrained to follow the path selected by qBGP and can also follow an alternate available QoS path. For selecting a path, the PCS can rely on the number of domain hops and/or on the QoS performances of each corresponding e-QC, or any other administrative local policy enforced.

Comparing e-QC is not an easy task. It is suggested that this comparison is achieved using the definition of the *Meta-QoS-Class* itself, which is supposed to particularly optimise one of the performance parameters of a QoS-Class (if a given *Meta-QoS-Class* has been defined for delay sensitive kind of application it can try to optimise its researches using this performance parameter). Thus, if can classify the learned paths according to this specific performance parameter and choose, from this perspective, the best egress point.

If the requested LSP is multi-coloured, it must select a path supporting all the requested *Meta-QoS-Classes*. If it finds more than one, the choice of the path can become much more tricky. When possible (e.g. when requested *Meta-QoS-Classes* belong to the same hierarchical branch), it is suggested to exploit the fact that *Meta-QoS-Classes* are hierarchically organized and to base the selection process on the highest *Meta-QoS-Class*. When Meta-QoS-Classes belong to different branches of the hierarchical tree there is no evident selection criteria. In such a case, it is suggested that the requesting PCC, indicates the priority order that will be used by PCS for searching a path.

If no path can be found the PCS must return an error. If a multi-coloured LSP was requested, one could imagine that the PCS could proceeds to an LSP splitting, providing it found different paths to reach the destination within the requested *Meta-QoS-Class* plane.

If an LSP (or one of the *Meta-QoS-Class* aggregated flows within the LSP) requests some bandwidth protection, the PCS can ignore the loss rate performance parameter of the corresponding *Meta-QoS-Class*. Indeed, considering bandwidth will be successfully provisioned for that LSP, no datagram loss will occur providing that the end-user respected the related service contact and doesn't sent more traffic for a given class that what was agreed in the cSLS.

7.3.3.5 QoS path establishment

When a PCS looks for a possible inter-domain QoS path, it will closely interact with its inter-domain management system. Indeed, PCS interactions will not only find a QoS path but will also verify that necessary resources are available and can be reserved. In order to achieve this goal, each PCS should have an accurate view of availability of the requested bandwidth:

- Along the path followed to cross its domain (intra-domain).
- At the inter-domain boundary the QoS path is supposed to use.

A given PCS can have several pending queries in progress. Resources requested by those queries will very likely interfere and simultaneously ask for common resources along the same path. Consequently, PCS must take care of that and must register in their management system a PRE-RESERVATION-INTENT of the corresponding resources. When PCRM are received, the state of each related resource should change to PRE-RESERVATION to indicate that the corresponding network resources will be engaged soon in an LSP set-up, or freed if a PCEM is received. Since the effective realization of the LSP is done via RSVP TE, there must be some very close interaction between PCS and RSVP TE mechanisms so that the distributed transaction can be monitored and error cases tracked (if the RSVP TE Path message is never sent for instance, PCS corresponding states should be freed). Each time an RSVP TE Resv crosses a domain boundary, some interaction between RSVP TE and PCS must occur (thanks to COPS for RSVP for instance) so that PCS can definitely register that resources have been effectively reserved and used.

The inter-working between the set of PCS and technical set-up mechanisms should be considered as a distributed transaction. LSP disassembling and breakdown should be considered in the same way.

This also means that each PCS will have to keep track of all individual inter-domain established LSP, which will be assigned a global and unique identifier.

7.3.3.6 Bandwidth reservation considerations

The basic MESCAL approach doesn't allow customers to request for bandwidth guarantees. But, as part of the QC-implementation process, the provider has, in some way, to allocate a given maximum

bandwidth for each of the *Meta-QoS-Classes* it supports: in its own network but also at its boundaries when pSLS have been established.

Figure 41: Bandwidth Repartition per MC

The figure above illustrates this bandwidth repartition using 3 *Meta-QoS-Classes* within this example. This provisioning must be achieved on all links of the domain and at each inter-domain peering edge. At the peering edge, these maximum bandwidths are those, which have been agreed in the pSLS. Within the network their values are at the discretion of the provider and reflect an optimal balancing of business and traffic engineering objectives.

In addition, the QoS MPLS TE end-to-end extension, introduces additional engineering issues which are depicted below:

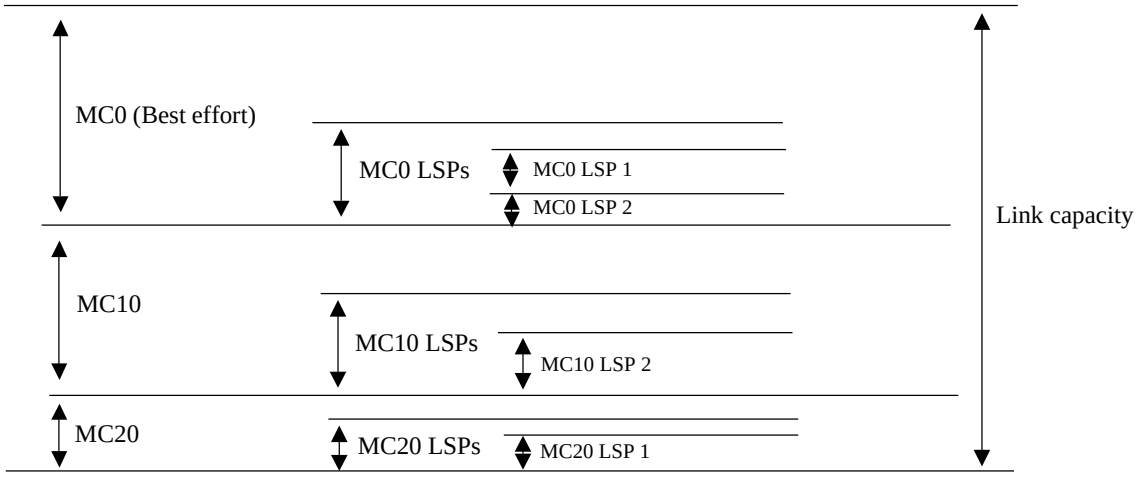


Figure 42: LSPs BW Reservation across multiple MCs

On a same link, for a given *Meta-QoS-Class*, both non-protected and protected traffics live together. These traffics can be IP traffic issued within the scope of the basic approach or within its extended scope (QoS MPLS TE). The traffic conditioning mechanism must be able to handle them in parallel in a consistent manner.

MCx LSPs, represents the maximum bandwidth which can be allocated to the MCx bandwidth protected traffic. Doing this prevents the non-protected traffic to fall into a starvation situation. MCx LSPs, represents also the amount of bandwidth that is always available to the MCx protected traffic. Depending on the level of control the provider has over its network, the maximum bandwidth that can be allocated to protected LSPs, should either be considered as an administrative maximum upper-bound or be enforced with appropriate mechanisms. Protected LSPs should be handled in such a way

they never experience any datagram loss. In fact, if traffic policing is correctly achieved at the ingress point of the LSP, no loss should be observed for bandwidth-protected traffic. The remaining bandwidth is used by the IP traffic. The minimum bandwidth, which can become available, is MCx bandwidth minus MCx LSPs bandwidth. When no LSPs are established, the non-protected IP traffic can potentially use all the MCx bandwidth.

7.3.3.7 QoS guarantees

With this end-to-end approach the guarantees provided are end-to-end QoS performances. In addition, if the LSP requests it, bandwidth guaranties can be provided.

It should be noted that in this latter case, loss rate is 0% when the end-user respects its traffic contract.

7.3.3.8 Terms of cSLS

The cSLS agreed between the provider and the end-user will have to specify:

- The destination edge of the traffic
- The maximum bandwidth of each LSP (Or per MC)
- The *Meta-QoS-Classes* requested, together with their maximum bandwidth and, for each of them:
  - The guaranteed bandwidth, if any is requested, with must be smaller or equal than the maximum requested bandwidth for the *Meta-QoS-Class*.
- The nature of the LSP (mono or multi-coloured)

7.3.3.9 Terms of pSLS

In addition to QoS contractual terms stated in the basic section of pSLS, this end-to-end approach leads to introduce specific contractual parameters.

- an agreement between the parties allowing the requestor to dynamically establish inter-domain LSP
- a list of possible destination restrictions (probably handled by the PCS of each domain)
- a specification of the maximum bandwidth dedicated for each *Meta-QoS-Class* (including the best-effort one)
- a specification of the maximum bandwidth dedicated to bandwidth guaranteed LSP, within each *Meta-QoS-Class*.
- plus any other appropriate clauses such as the maximum number of LSP that can be requested, or the maximum bandwidth per LSP…

7.3.3.10 On demand inter-domain pSLS interactions

When computing a path, the PCS can fail for intra-domain and/or inter-domain reasons. Those failures, in normal operations, will be mainly due to the lack of resources. In such a situation, a path, which would have been the optimal path, would not be established. Identification of the domain where the path computation failed, together with the associated reasons, would be of a real added value for providers in order to improve the service they offer, thanks to an appropriate remote pSLS (re) negotiation request.

One way for achieving that is to rely on the Path Computation Protocol, which could be improved to return all the path alternatives which were tried but which failed. Doing so, the requesting provider would be aware of the reasons of the failure and possibly interact with the remote failing AS.

The remote AS, confronted to multiple requests, from external domain, could objectively consider a possible modification of some of its pSLSs based on objective business incitements.

7.3.3.11 IPv6 support

No major specific IPv6 issues were raised excepted MPLS TE support within an IPv6 environment.

7.3.3.12 Scalability

Clearly, if this solution option were deployed for all Internet users, it would not be scalable at all. But, this solution option has been designed to support the hard guarantees service option, which is mainly dedicated for mission critical applications, and so, targets corporate users and/or added-value services providers. Since the solution effectively reserves appropriate network resources across multiple-domains, providers pricing policies will be consequently adapted and will naturally regulate the usage of this service option. It will be deployed only when the interested future/potential customers will show clear demands. This is the reason why it is not expected that a large number of inter-domain LSPs be deployed, which would lead to non-scalable deployments in terms of number of LSP to be maintained and engineered. No full-mesh of LSPs is expected nor considered.

In each domain, the number of requests each PCS will have to answer will consequently be limited and thus, PCS systems should treat a predictable and a reasonable number of requests. Path computation is made easier thanks to the use of qBGP, which advertises the QoS performance associated with each selected path. The number of inter-PCS queries might become important when the bandwidth criteria cannot be satisfied (note that this is classical client/server behaviour and no over computing is added), but this could be improved using some specific qBGP extensions reporting the available bandwidth which can still be reserved toward a destination.

7.3.3.13 Applicability to Business Model

Thanks to the QoS MPLS TE extension, corporate business can be targeted:

- QoS performances become stable because the path followed by a given LSP is now fixed.
- Bandwidth guarantees can be offered, because it becomes possible for the provider to allocate appropriate resources (and no more) all along the path followed by the LSP.

In this option, pSLS becomes very strict and each crossed provider's domain must commit to respect all terms of the pSLS, since an LSPs have an end-to-end meaning.

7.3.4 Multicast support

7.3.4.1 Service Model Selection

The QoS support for multicast services in the MESCAL project will be based on the Source Specific Multicast (*SSM*) service model, and the reason being the following:

- Market requirement. It has been realised that single source applications (e.g., Internet TV, content distribution etc.) are currently the major driving force of developing multicast services [Diot00] across the Internet. In these applications, the group source is either already well-known or it can be obtained by offline approaches, and no additional mechanisms need to be appended to the existing framework. The SSM service model was proposed for efficient support to this type of services.
- Scalability. Since each group session is identified with both source address and class D address (in 232/8), Inter-domain group address allocation will not become an issue. Moreover, explicit group join from individual receivers eliminates the necessity of employing source discovery mechanisms (e.g., MDSP), which still suffer from problems (security, scalability etc) in the deployment across the Internet.
- Practical implementation. Although BGMP/MASC has been reckoned as the long-term solution for multicast services in the future, and it also fits in with the cascaded QoS models in theory, it has not seen any significant practical progress and still remains in its research stage. This lacking

of popularity and support in practice (e.g., products from vendors) and this is one of the risks to consider QoS extensions to BGMP/MASC. In effect, the cascaded QoS service model can also work well in the SSM paradigm, and we will illustrate the basic mechanism in the following sections.

7.3.4.2 Design of multicast SLS

We observe that almost all of the multicast services are receiver initiated, and it is group members that demand heterogeneous QoS requirements based on individual capacities. Moreover, according to the concept of IP multicast and SSM model, group members are always anonymous to the multicast source. These characteristics make it difficult to apply sender based SLS to multicast services, and hence a set of dedicated multicast SLS is required. In the same fashion to the unicast scenario, we classify multicast SLS into customer SLS (mcSLS) and provider SLS (mpSLS). The fundamental difference between the unicast cSLS and multicast cSLS (mcSLS) is as follows: In the unicast scenario, the cSLS subscriber (i.e., unicast source) negotiates with its ISP on the *e-QC* by which its data can be sent to a specific destination prefix in the Internet. Based on the pSLS between transit domains, this cSLS will return a set of destination prefixes on per e-QC basis. In Figure 43(a), it can be inferred from the cSLS that the sender *s* attached to AS0 may have its data delivered to a destination *r1* with e-QC1, e-QC2 and e-QC3, since *r1* is located in the common area of the three e-QC zones. However, data from the sender to the destination *r2* can be only delivered with e-QC3 since none of the rest e-QCs is able to span their zones to this destination prefix. In the multicast scenario, the mcSLS subscriber should be a receiver other than the sender, and the negotiation with its ISP is on the e-QC by which this group member can receive data from a remote source prefix. Figure 43(b) presents an example of the mcSLS negotiation. A particular group member *r* attached to AS0 may receive multicast group data from the source *s1* based on e-QC1, e-QC2 and e-QC3, while it can only receive data from *s2* by e-QC3 since *s2* belongs to the source prefix that is only covered by the e-QC3 zone. For a particular receiver, before it sends the group join request towards the source s, it should first negotiate with the ISP through mcSLS on whether s can be reached based on the requested e-QC. If not, the receiver will select an alternative e-QC that is able to span to the source s. In this scenario, when s receives a group join with demanded QoS, it need not care whether or not the remote receiver can be reached based on this requirement.

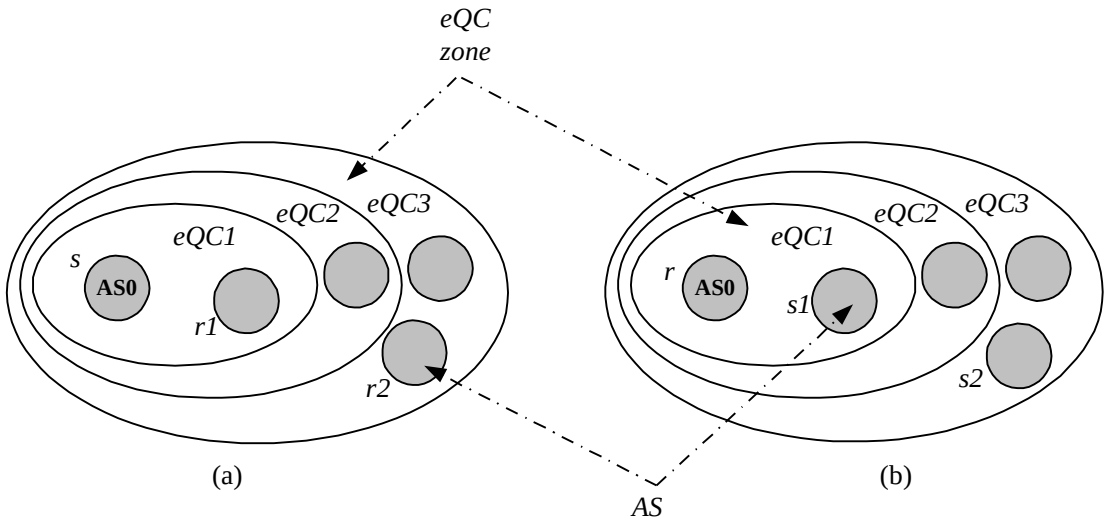


Figure 43: e-QC based SLS

Similar to the multicast cSLS scenario, peering ISPs should also set up dedicated pSLS for multicast traffic aggregates on per *e-QC* basis. In contrast to the unicast pSLS, mpSLS is receiver-oriented in that the requesting AS (i.e., downstream AS) negotiates with its peering on the *e-QC* by which multicast aggregate can be delivered from upstream source prefix to the local AS. After performing the qc binding, this local AS will be able to offer a set of *e-QCs* that can be referred by both local

McSLSs and further downstream mpSLSs. Figure 44 presents a basic illustration on mpSLS set-up between two adjacent domains AS1 and AS2. In this example, AS1 requests to AS2 on mpSLS regarding *e-QCs* from a source prefix s in the remote AS X. It should be noted that this mpSLS concerns the uni-directional QoS parameters from s back to AS1 (which is in the opposite direction of unicast pSLS), since mcSLS requires that corresponding multicast traffic flow in this direction. After qc binding between AS1 and AS2, AS1 will be able to negotiate mcSLS with its local receiver (e.g., *r1* and *r2*). Moreover, AS1 is also able to offer an extended *e-QC* to the downstream AS (AS0 in this example) for further mpSLS negotiations.

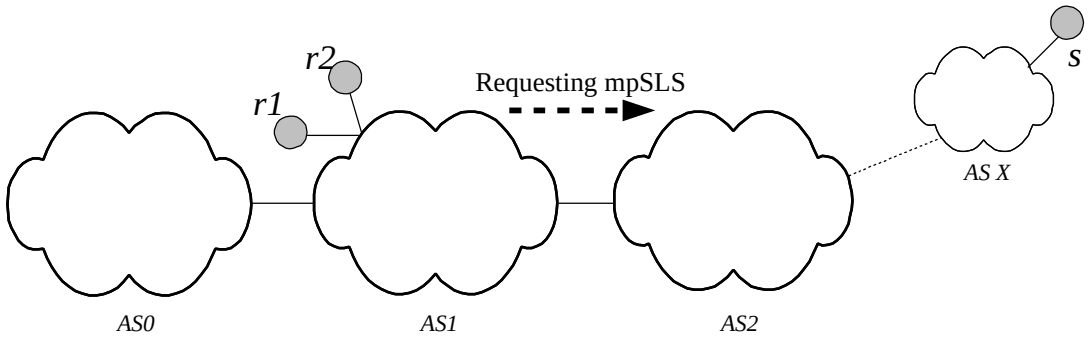


Figure 44: Multicast pSLS (mpSLS)

7.3.4.3 Design of Multicast routing

To enable QoS aware multicast services across the Internet, multicast trees with QoS requirements are constructed. As far as DiffServ environment is concerned, there exist two strategies in building multicast trees: per QoS class trees and hybrid trees. In the first approach, multicast trees are QoS class specific, i.e., within a group session one dedicated tree serves each particular QC. If a multicast application uses k QCs, there will be k independent trees. In the hybrid approach, only one single tree is constructed for each group session, and this unique tree contains multiple classes of services, with individual tree branches reflecting heterogeneous QoS requirements. The key idea of this type of tree is that branches with lower classes can be directly grafted from those with higher classes for the same group. The advantage of per QC trees lies in its simplicity in implementation, while the hybrid tree approach has its virtue in bandwidth and group state conservation. The Mescal project will consider both of the two strategies.

From a viewpoint of routing implementation, we will also consider the following two directions. (1) Pure IP level routing with PIM-SM/MBGP, and (2) Fixed path with MPLS.

7.3.4.3.1 The PIM-SM/MBGP approach

The task of intra-domain multicast routing in the MESCAL project is to explore a feasible path that satisfies QC requirements between ingress and egress routers within one AS. Given the fact that PIM-SM has become the most popular multicast routing protocol, we are not going to invent another brand new solution that is completely incompatible with the underlying infrastructure. On the other hand, current PIM-SM is not a QoS-aware protocol, and hence sophisticated IP-layer network dimensioning (e.g., setting proper link weights to influence PIM-SM path selection) is required to achieve end-to-end QoS demands. In implementation, a dedicated QoS aware multicast routing information base (MRIB) is constructed for PIM-SM join requests to explore a feasible path that satisfies the required QoS demands. As we have previously mentioned, the conventional Reverse Path Forwarding (*RPF*) scheme cannot directly support QoS multicast routing. This is because if a multicast packet is not coming from the shortest path leading towards the source, it will always be discarded. In this sense, in order to prevent loops in PIM-SM routing, we will also investigate an adapted RPF checking based on the QoS enabled MRIB.

Although SSM eliminates the need for inter-domain source discovery, an inter-domain routing protocol is highly recommended for advertising the source NLRI information to other domains.

Currently MBGP is the *de facto* routing protocol for inter-domain *SSM* deployment. The task of MBGP is to advertise NLRI information for other protocols than IPv4 and other address formats than those of IPv4 unicast addresses; MBGP introduce two new attributes: MP_REACH_NLRI and MP_UNREACH_NLRI, and the UPDATE messages never carry multicast group addresses. MBGP is able to carry incongruent routes for unicast and multicast route by using different Subsequent Address Family Identifiers (*SAFI*) in the attributes of MP_(UN)REACH_NLRI. In order to achieve end-to-end QoS guarantee, MBGP needs to be extended for QoS-awareness, and we name it QoS-MBGP (qMBGP). The basic task of qMBGP is to advertise QoS parameters in MP_REACH_NLRI for multicast source reachability. Compared with qBGP for unicast traffic, its multicast functionality is on the QoS conditions *from* the source prefix. In Figure 45, the virtual lines identify logic connectivity between edge routers inside a domain while the solid lines indicate that external peering edge routers are physically connected. In order to advertise the QoS reachability of the source s, AS2 will first explore a feasible path that satisfies multicast QC requirement. We suppose that the uni-directional path R23→R22 meets this criterion, and then R22 will send to R13 a BGP UPDATE regarding source prefix s with SAFI equal to 2. On receiving the message, AS1 starts building its own e-QCs by QC mapping and binding. We assume that the path R13→R11 can satisfy the multicast e-QC and hence R11 will advertise a new bgp UPDATE with SAFI equal to 2 to its external peer R01. As a result, R01 will be able to send group join request towards s via the inter-domain path R01→R11→R13→R22→R23, and we can find its reverse path is exactly what has been explored to satisfy the multicast e-QC from the source s. In summary, the function of qMBGP is to select edge routers (egress nodes) for further delivery of join request to build up a multicast tree branch that guarantees multicast e-QC, whereas it is the task of PIM-SM to decide the actual route between these selected internal peers with dynamic group membership.

This type of layer 3 routing paradigm is targeting at both loose service option and statistical service option. Particularly, in order to support the statistical service option, sophisticated network dimension for multicast traffic is needed.

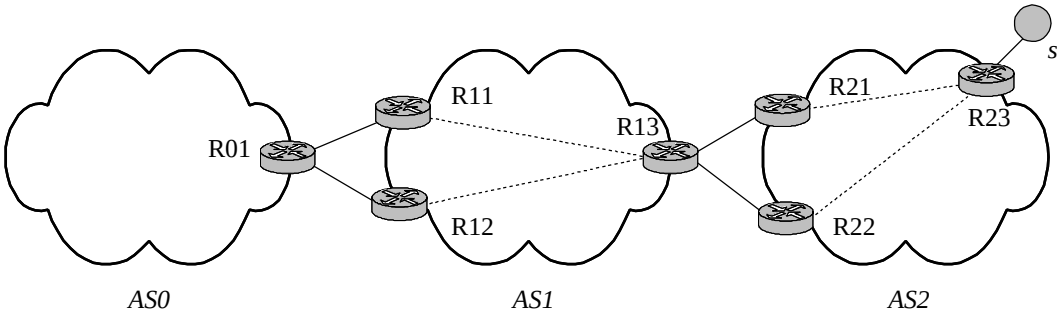


Figure 45: qMBGP path selection

7.3.4.3.2 The fixed path approach

Similar to its counterpart in unicast routing, the fixed path approach for multicast traffic also requires that the multicast tree should be fixed. In this sense, once the multicast tree is computed that meets the required QoS guarantees, all the forthcoming multicast traffic will flow on the fixed tree from the source to individual group members. The advantage of this approach is that there exists higher flexibility in routing decision considering end-to-end QoS requirements. For example, QoS aware Steiner trees become possible, while they meet difficulties in PIM-SM that only supports shortest path routing. If MPLS is adopted for QoS aware explicit routing, there is one significant difference between the unicast and multicast scenario. As we have previously mentioned, both mcSLS and mpSLS are receiver initiated, and this requires that the corresponding routing should also be group member oriented, which exactly conforms to the conventional IP multicast and SSM model. In implementation, the computed fixed path is used for delivery of individual group join requests, other than multicast traffic itself. The strategy of selecting each join path is based on its reversed QoS parameters, i.e., the path for delivery of join requests must meet the following request: the corresponding reversed QoS

condition along this path must satisfy the associated mcSLS/mpSLS. On the other hand, the join request packets need not receive any QoS treatment and they can be delivered towards the source on the Best Effort basis.

This type of fixed path paradigm targets the statistical service option and hard service option. Especially, efforts will be made towards the achievement of the latter option by means of the MPLS mechanism, which is still an open issue.

8 REFERENCES

[Awduc02] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, *Overview and Principles of Internet Traffic Engineering*, IETF RFC3272, May 2002

[Bates00] T. Bates et al, *Multiprotocol Extensions to BGP4*, RFC 2858, June 2000

[Cain02] B. Cain et al, *Internet Group Management Protocol, Version 3*, RFC 3376, Oct. 2002

[Crist02] G. Cristallo, C. Jacquenet, *Providing Quality of Service Indication by the BGP-4 Protocol: the QOS_NLRI attribute*, IETF Internet draft, <draft-jacquenet-qos-nlri-04.txt>, 2002

[Deeri88] S. Deering, *Multicast Routing in Internetworks and Extended LANs*, Proc. *ACM SIGCOMM*, 1988, pp55-64

[Diot00] C. Diot *et al*, *Deployment Issues for the IP Multicast Service and Architecture*, *IEEE* Network, Jan./Feb. 2000, pp 78-88

[Feldm00] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, and J. Rexford, *Netscope: Traffic Engineering for IP Network*", IEEE Network Magazine, special issue on Internet traffic engineering, March/April 2000, pp. 11-19

[Fenne03a] B. Fenner et al, *Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol specification (Revised*", draft-ietf-pim-sm-v2-new-07.txt, work in progress, 2 Mar 2003

[Fenner03b] B. Fenner et al, *Multicast Source Discovery Protocol (MSDP)*, draft-ietf-msdp-spec-20.txt, work in progress 2000

[Fortz00] B. Fortz, M. Thorup, *Internet Traffic Engineering by Optimizing OSPF Weights*, In Proc. INFOCOM 2000

[Goder02] D. Goderis et al., *Service Level Specification Semantics, Parameters and Negotiation Requirements*, IETF Internet draft, draft-tequila-sls-02.txt, February 2002

[Goder02a] D. Goderis et al., *A Service-Centric Quality of Service Architecture for IP-based Next Generation Networks*, Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS'02), Florence, Italy, R. Stadler, M. Ulema, eds., pp. 139-154, IEEE, April 2002

[Holbr03] H. Holbrook et al, *Source-Specific Multicast for IP*, draft-ietf-ssm-arch-03.txt, work in progress, 7 May 2003

[Huston99] Geoff Huston, *Interconnection, Peering and Settlements-Part II*, The Internet Protocol Journal, Cisco Publications, vol 2, no 2, June 1999.

[Lefau03a] F. Le Faucher, W. Lai, *Requirements for support of Diff-Serv-aware MPLS Traffic Engineering*, IETF Internet draft, draft-ietf-tewg-diff-te-reqts-07.txt, August 2003

[Lefau03b] F. Le Faucher, *Russian Dolls Bandwidth Constraints Model for Diff-Serv-aware MPLS Traffic Engineering*, IETF Internet draft, draft-ietf-tewg-diff-te-russian-02.txt, September 2003

[Lefau03c] F. Le Faucher, *Maximum Allocation Bandwidth Constraints Model for Diff-Serv-aware MPLS Traffic Engineering*, IETF Internet draft, draft-lefaucheur-diff-te-mam-00.txt, February 2003

[Nichols01] K. Nichols, and B. Carpenter, *Definition of Differentiated Services Per Domain Behaviors and the Rules for their Specification*, IETF RFC3086, April 2001

[Rados00] P. Radoslavov et al, *The Multicast Address-Set Claim (MASC) Protocol*, RFC 2909, Sept. 2000

[Rekht95] Y. Rekhter, T. Li (eds.), *A Border Gateway Protocol 4 (BGP-4)*, IETF RFC, Standards Track, March 1995

[RFC2460] Deering, S., Hinden, R., *Internet Protocol, Version 6 (IPv6) Specification*, RFC 2460, Standards Track, December 1998

[RFC2474] Nichols, K., Blake, S., Baker, F., *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*, RFC 2474, Standards Track, December 1998

[TEQUI] IST-TEQUILA Project, information available at www.ist-tequila.org

[Thale03] D. Thaler et al, *Border Gateway Multicast Protocol (BGMP): Protocol Specification*, draft-ietf-bgmp-spec-05.txt, work in progress

[Trimin01] P. Trimintzios, I. Andrikopoulos, G. Pavlou, P. Flegkas, D. Griffin, P. Georgatsos, D. Goderis, Y. T'Joens, L. Georgiadis, C. Jacquenet, R. Egan, *A Management and Control Architecture for Providing IP Differentiated Services in MPLS-based Networks*, IEEE Communications, special issue in IP-Oriented Operations and Management, Vol. 39, No. 5, pp. 80-88, IEEE, May 2001

[Trimin03] P. Trimintzios, G. Pavlou, P. Flegkas, P. Georgatsos, A. Asgari, E. Mykoniati, *Service-driven Traffic Engineering for Intra-domain Quality of Service Management*, IEEE Network, special issue on Network Management of Multi-service, Multimedia, IP-based Networks, Vol. 17, No. 3, IEEE, May/June 2003

[Vasse03] Vasseur, JP., Zhang, R., *Inter-AS MPLS Traffic Engineering*, draft-vasseur-inter-as-te-00.txt, February 2003

[Walton02] D. Walton, D. Cook, A. Retana, J. Scudder "Advertisement of Multiple Paths in BGP" IETF Internet Draft, <draft-walton-bgp-add-paths-01.txt>, November 2002

[Zhang03] Raymond Zhang, Editor (Infonet Services Corporation), JP Vasseur (Cisco Systems), *MPLS Inter-AS Traffic Engineering requirements*, Internet draft: draft-zhang-mpls-interas-te-req-01.txt, Expires: July 2003.

9 ABBREVIATIONS

AS	Autonomous System
ASO	AS Origin
ASS	AS Sibling
CAS	Central AS
cSLS	Customer SLS
DAS	Destination AS
DIFFSERV	Differentiated Services
DSCP	Differentiated Services Code Point
e-QC	Extended-QC
FIB	Forwarding Information Base
g-QC	Global-QC
IGMP	Internet Group Management Protocol
IPA	IP Address
ISP	Internet Service Provider
LL	Local Label
LST	Label Switching Table
MC	Meta-QoS-Class
m-QC	measured-QC
NRS	Neglected Reservation Sub-Tree
PIM-SM	Protocol Independent Multicast Sparse Mode
pSLS	Provider SLS
QC	QoS Class
QoS	Quality of Service
RPF	Reverse Forwarding Path
RIB	Routing Information Base
RL	Remote Label
SLA	Service Level Agreement
SLS	Service Level Specification
SSM	Source Specific Multicast
TAS	Transit AS

10 APPENDIX A: STATE OF THE ART REVIEW OF RESEARCH, STANDARDISATION AND CURRENT COMMERCIAL PRACTICE IN INTER-DOMAIN QOS DELIVERY

10.1 Introduction

The overall objective of the MESCAL project is to propose and validate scalable, incremental solutions that enable the flexible deployment and delivery of inter-domain Quality of Service (QoS) across the Internet. MESCAL will validate its results through prototypes, and evaluate the overall performance through simulations and prototype testing. MESCAL will contribute to standardization efforts, especially those conducted by the Internet Engineering Task Force (IETF), participate in IST clustering and actively disseminate its results.

This document is part of Work Package 1, "Specification of Functional Architecture, Algorithms and Protocols." It provides a summary of the current status of standards, research work and commercial implementations of technologies that enable or support inter-domain and intra-domain QoS.

The structure of the document is as follows:

- Section 2 reviews the current status of the IETF Differentiated Services (DiffServ) initiative, and gives a taxonomy of QoS-sensitive applications, describing their QoS requirements in terms of packet loss rate, delay, jitter and throughput.
- Section 3 considers intra-domain traffic engineering (TE), describing a number of proposals that add TE capabilities to IGPs such as OSPF.
- Section 4 considers inter-domain TE, discussing BGP- and MPLS-based mechanisms, and reviewing a number of research approaches for introducing inter-domain QoS using each of these two mechanisms.
- Section 5 reviews signalling protocols (BGRP, SIBBS, RSVP and SIP).
- Section 6 reviews two service management models. The first is the Internet QoS service model, looking at the two frameworks IntServ and DiffServ; and the second is the TEQUILA QoS service model framework.
- Section 7 considers admission control schemes to support QoS enforcement, including bandwidth brokers, endpoint admission control, distributed admission control, and end-point admission control (probing). The approach adopted in TEQUILA is also reviewed.
- Section 8 reviews multicast, including a review of intra- and inter-domain non-QoS multicast routing protocols, QoS multicast routing protocols, and issues of multicast in DiffServ networks.
- Section 9 reviews the impact of IPv6 on QoS services, specifically the traffic class field, flow labels and extension headers.
- Finally, Section 10 reviews the state-of-the-art in policy-based networking, considering research algorithms and IETF work and the COPS framework including QoS-related and TE-related proposals.

10.2 DiffServ Update, Traffic and Applications

10.2.1 DiffServ Update

With the publication of the DiffServ PIB [RFC 3317], the DiffServ charter was completed and the working group was officially closed in March 2003. The working group was very successful,

producing 15 RFCs (listed in Table 1) and the technology that is widely seen as a key part of the Internet toolboxes.

RFC	Title
RFC 2474	Definition of DiffServ field (DS field) in the IPv4 and IPv6 headers
RFC 2475	An Architecture for DiffServ
RFC 2597	Assured Forwarding PHB Group
RFC 2983	DiffServ and tunnels
RFC 3086	Definition of DiffServ per Domain Behaviours and rules for their specification
RFC 3140	Per Hop Behaviour Identification Codes. RFC 2836 obsoleted by this RFC.
RFC 3246	An Expedited Forwarding PHB. RFC 2598 obsoleted by this RFC.
RFC 3247	Supplemental Information for the New Definition of EF PHB.
RFC 3248	A delay Bound alternative revision of RFC 2598.
RFC 3260	New Terminology and Clarification for DiffServ.
RFC 3289	Management Information Base for the DiffServ Architecture
RFC 3290	An Informal Management Model for DiffServ Routers
RFC 3317	Differentiated Services Quality of Service Policy information Base

Table 3: Produced RFCs by DiffServ Working Group.

Differentiated Services allows an approach to IP Quality of Service (QoS) that is modular, incrementally deployable, and scalable while introducing minimal per-node complexity [RFC2475]. From the user's point of view, QoS should be supported end-to-end between any pair of hosts. However, this goal is not immediately attainable with current technologies: it requires inter-domain QoS support, which is an on-going research issue. A goal of the DiffServ WG was to provide the firm technical foundation that allows the business models for inter-domain QoS to be developed outside of the IETF. The first major step was to support edge-to-edge or intra-domain QoS between the ingress and egress of a single network, i.e., a DS Domain in the terminology of RFC 2474. The intention was that this edge-to-edge QoS should be composable, in a purely technical sense, to a quantifiable QoS across a DS Region composed of multiple DS domains.
The DiffServ WG standardised the behaviours required in the forwarding path of all network nodes, the per-hop forwarding behaviours or PHBs. The PHBs defined in RFCs 2474, 2597 and 2598 give a rich toolbox for differential packet handling by individual boxes. The general architectural model for DiffServ has been documented in RFC 2475. An informal router model describes a model of traffic conditioning and other forwarding behaviours.

10.2.1.1 Framework and Architecture

The DiffServ architecture is designed to be scalable in terms of network size and speed, by keeping all complexity at the network edges. It keeps any per-flow state information and performs all complicated per-flow packet processing (e.g. shaping, policing) at the network edges. The QoS service 'signalling' is explicitly carried in the IP datagram header using the original “Type of service filed” renamed as the DiffServ field [RFC2474]. This provides high-speed forwarding in the core of the network.

Traffic is categorised into a limited number of classes. DiffServ Code points are defined for this categorisation although some limited compatibility with the precedence notion in the ToS field is preserved. The DSCP is used for QoS forwarding.

The IETF consciously decided not to standardise services *per se*, but rather to specify only particular router forwarding behaviours, known as Per Hop Behaviours (PHB). These are intended to allow Internet service providers complete freedom to construct, from PHBs, the intra-domain services they

believe will meet their customers' needs. DS field marking will typically be performed only once, in the user network or at DS network boundary, thereby marking each packet for a specific PHB according to a pre-arranged service level specification. Router resources (bandwidth and possibly buffer space) are allocated to each supported PHB according to service provisioning policies.

10.2.1.2 Defined Per-Hop Behaviours (PHBs)

Two PHBs have been specified, and the codepoint '000000' is explicitly reserved for best effort traffic.

- The first PHB, called *Expedited Forwarding* [RFC-2598], is aimed at creating services with *virtual leased line*-characteristics (these are called *Premium Services* in [RFC2638]). This means, that the connection should behave as if it was running on its own dedicated link. To put it in more QoS-like terms: the connection should experience an end-to-end low delay, low jitter and assured bandwidth.
- A second class of PHBs has been named *Assured Forwarding* [RFC2597] (referred to as the *Olympic services* in [RFC-2638]). The traffic in such a class will have at least a given probability to be forwarded (as long as it stays within the agreed profile), but it can be placed in a buffer (this means: nothing can be said about delay or jitter). If an AF-class is over its limit and the corresponding queue is getting congested, there is the choice between remarking and dropping packets. For this reason, four AF-classes are defined, each with three drop-probabilities. This way a weighted dropping-behaviour can be implemented. Furthermore, a DiffServ node should not reorder IP-packets of the same microflow if they belong to the same AF-class.

10.2.1.3 DiffServ Routers

One of the major strengths of DiffServ is its scalability. This is obtained by moving the complexity of the architecture towards the edge (where the packet-density is lower) and keeping the core as simple as possible. This also means that two types of models must be given for the routers in a DiffServ domain. A core router in the DiffServ-network applies the right PHB to the incoming IP-packets based on their DSCP and forwards the packets. This is all done on behaviour aggregates, so a scalable solution is obtained.

A boundary router on the other hand performs classification and traffic conditioning (TC) in addition to supporting the functionality of a core router. Four traffic conditioning functions are defined in [RFC2475]: metering, marking, shaping and policing/dropping. Classification is based on a DSCP (for host marked traffic) or a certain set of fields.

10.2.1.4 Per Domain Behaviours (PDBs)

The goal of creating scalable end-to-end QoS in the Internet requires that we can identify and quantify behaviour for a group of packets that is preserved when they are aggregated with other packets as they traverse the Internet. RFC 3086 defines and specifies the term "Per-Domain Behaviour" (PDB) to describe QoS attributes across a DS domain. PDB is defined the expected treatment that an identifiable or target group of packets will receive from "edge-to-edge" of a DS domain. A particular PHB (or, if applicable, list of PHBs) and traffic conditioning requirements are associated with each PDB.

DiffServ classification and traffic conditioning are applied to packets arriving at the boundary of a DS domain to impose restrictions on the composition of the resultant traffic aggregates, as distinguished by the DSCP marking, inside the domain. The classifiers and traffic conditioners are set to reflect the policy and traffic goals for that domain and may be specified in a Traffic Conditioning Agreement (TCA). Once packets have crossed the DS boundary, adherence to DiffServ principles makes it possible to group packets solely according to the behaviour they receive at each hop (as selected by the DSCP). This approach has well-known scaling advantages, both in the forwarding path and in the control plane. The PHB must be equivalent for every node in the domain, while the set of packets marked for that PHB may be different at every node. PHBs should be defined such that their characteristics do not depend on the traffic volume of the associated BA on a router's ingress link nor

on a particular path through the DS domain taken by the packets. Specifically, different streams of traffic that belong to the same traffic aggregate merge and split as they traverse the network. If the properties of a PDB using a particular PHB hold regardless of how the temporal characteristics of the marked traffic aggregate change as it traverses the domain, then that PDB scales. Clearly this assumes that numerical parameters such as bandwidth allocated to the particular PDB may be different at different points in the network, and may be adjusted dynamically as traffic volume varies.

There is a clear distinction between the definition of a Per-Domain Behaviour in a DS domain and a service that might be specified in a Service Level Agreement. The PDB definition is a technical building block that permits the coupling of classifiers, traffic conditioners, specific PHBs, and particular configurations with a resulting set of specific observable attributes, which may be characterised in a variety of ways. These definitions are intended to be useful tools in configuring DS domains, but the PDB/s used by a provider is not expected to be visible to customers any more than the specific PHBs employed in the provider's network would be. Network providers are expected to select their own measures to make customer-visible in contracts and these may be stated quite differently from the technical attributes specified in a PDB definition, though the configuration of a PDB might be taken from a Service Level Specification (SLS). Similarly, specific PDBs are intended as tools for ISPs to construct differentiated services offerings; each may choose different sets of tools, or even develop their own, in order to achieve particular externally observable metrics. Nevertheless, the measurable parameters of a PDB are expected to be among the parameters cited directly or indirectly in the Service Level Specification component of a corresponding Service Level Agreement (SLA).

10.2.1.5 Policy Information Base (PIB)

RFC 3317 was the last RFC published by the DiffServ WG: it describes a Policy Information Base (PIB) for a device implementing the DiffServ architecture. The defined provisioning classes provide policy control over resources implementing the DiffServ Architecture. These provisioning classes can be used with other none-DiffServ provisioning classes (defined in other PIBs) to provide for a comprehensive policy controlled mapping of service requirement to device resource capability and usage.

10.2.1.6 A Layered Service Model for DiffServ: TEQUILA view

One of the basic DiffServ QoS concepts is the PHB, exposing, in a generic way, the QoS capabilities of a router. PHBs may be implemented by a range of scheduling and buffering mechanisms such as Priority Queuing, Class Based Weighted Fair Queuing (CB-WFQ) and algorithms for implementing packet-dropping policies such as Weighted Random Early Detection (WRED). The PHB is the basic building block for supporting value-added IP services, previously negotiated between the provider and its customers through SLAs. Figure 46 shows a layered view of DiffServ service model from high-level IP transport service to the low-level data-plane concept of a PHB.

The upper two layers of the service model (Figure 46) describe the interface between the IP transport provider and the customer. According to the IETF DiffServ working group, a SLA is "*the documented result of a negotiation between a customer and a provider of an IP service that specifies the levels of availability, serviceability, performance, operation or other attributes of the transport service*". The SLA contains technical and non-technical terms and conditions. The technical specification of the IP connectivity service is given in SLSs. A SLS "*is a set of technical parameters and their values, which together define the IP service, offered to a traffic stream by a DiffServ domain*". SLSs describe the traffic characteristics of IP flows and the QoS guarantees offered by the network to these flows. Note that a SLA may contain a set of SLSs.

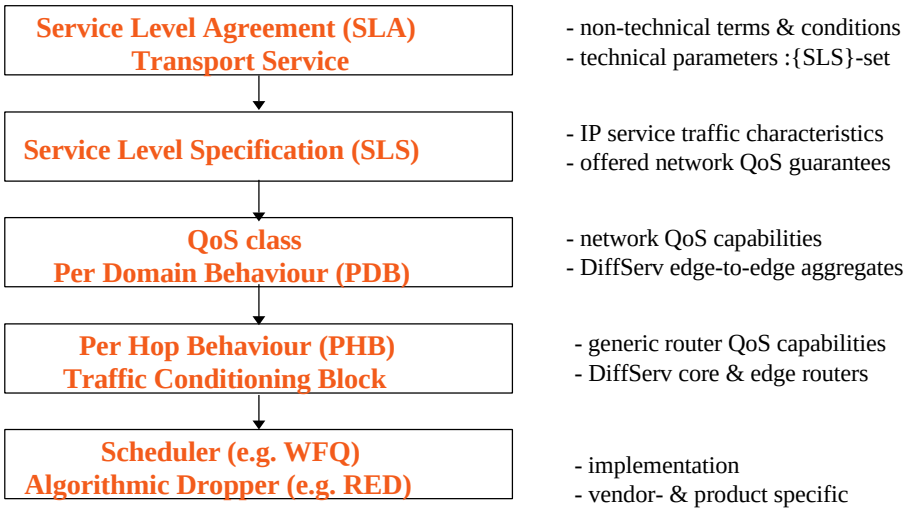


Figure 46: A proposal for a DiffServ Layered Service Model

10.2.1.7 Service Level Specifications (SLSs)

The DiffServ working group did not intend to specify further the content of a SLS beyond the loose definitions given above. Nevertheless, the definition of a SLS is a key-step towards the provisioning of value-added IP services because it specifies the semantics of the interface between the provider and the customer, i.e. *the technical terms and conditions*. To this end, TEQUILA project proposed a well-defined template for the parameters and semantics of SLSs describing the technical characteristics of QoS-based IP connectivity services. The basic parameter groups of the SLS template with a brief description of each are presented in Table 4. The *TEQUILA SLS template* describes the technical characteristics (topology, IP flows, transfer quality characteristics, compliance criteria) of a single, unidirectional 'connectivity leg'. A connectivity service then, is a collection of such SLS templates, bound to the same customer and the same access, usage means and characteristics. TEQUILA's definition of a SLS is uni-directional, thus requiring two symmetric SLSs to describe services such as a bi-directional Virtual Leased Line or a telephone call.

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

Table 4: SLS parameters and description.

10.2.1.8 QoS classes and PDBs

In establishing a holistic view for QoS service provisioning, there was a missing link between the concept of PHB, the basic QoS building block in IP DiffServ, and QoS-based services. The third layer in Figure 46 is the layer mediating between the customer-specific SLS-based services and the elementary PHBs supported by the routers. TEQUILA has filled this gap by introducing the notion of QoS class. The notion of the QoS class is introduced to substantiate this mediation. A QoS class consists of an Ordered Aggregate (or PHB Scheduling Class, e.g. AF1) and associated QoS parameter(s) such as one-way transit delay, inter-packet delay variation or packet loss (see Table 5).

QoS classes depict the elementary QoS transfer capabilities of a DS domain, between ISP edges. They are not services *per se*; instead services are built on them. QoS classes expose the network-wide QoS transport capabilities and they are bound to the specific network technology employed and capabilities provided by the network. For example, a Virtual Wire QoS class could be defined to denote an edge-to-edge transport capability with a guaranteed maximum packet delay and a guaranteed throughput for an aggregate IP packet stream. QoS classes should be seen as the PDBs. TEQUILA adopted the following definition of a QoS class.

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

Table 5: Definition of a DiffServ QoS class.

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

A network supports certain QoS classes through deploying dedicated TC blocks at the edge routers, PHBs throughout the network, and an overall resource management system that includes Bandwidth Broker like capabilities. The need for BB capabilities was identified in RFC-2638.

10.2.2 Applications

10.2.2.1 Transport Protocols Used by Applications

The transport layer is responsible for hiding the workings of the network layer and providing a means for an "application" (i.e. transport layer client) to communicate with its peer on another node in the network with a defined quality of service. There are two types of transport layer: connection-oriented and connectionless. The predominant transport protocols are TCP and UDP, both developed by the IETF. Two other recently-developed transport layer protocols described below are SCTP and DCCP.

TCP: TCP provides a connection-oriented byte stream service. The transmitter segments the data provided by the client according to the receiver's capacity to receive. TCP is used by elastic applications. RFC1006 specifies how TCP can be used as the transport layer for applications expecting an ISO OSI Reference Model Transport Layer.

UDP: UDP provides a packet-based connectionless transport service; it is a layer above IP adding error detection and multiplexing functionality. Being connectionless, any client must handle packet loss and packet re-ordering. UDP is used by inelastic applications.

SCTP: SCTP is targeted for signalling (e.g., audio/video signalling over IP). In early 1999, multiple technology vendors established the IETF SIGTRAN Working Group, the industry organisation responsible for developing and standardising protocols for the transport of packet-based mobile/PSTN signalling over IP networks. In October 2000, the group officially adopted the Stream Control Transmission Protocol (SCTP, RFC 2960) as the base protocol for SS7 over IP (SS7oIP). SCTP tries to address the perceived "deficiencies" of TCP. Among these are the fact that TCP presents a byte stream service, which is not suitable for, for example, users wishing to timestamp their data packets, whereas SCTP presents a packet based service. The result is a robust session-layer protocol that ensures retransmission and reliable end-to-end delivery of packets in the event of backbone congestion. The IETF has also worked in draft form on a series of adaptation layers on top of SCTP that will enable services such as Message Transfer Part-User-Peer-to-Peer Adaptation Layer (M2PA). However, development of this protocol has begun and it has yet to make an impression in Commercial off The Shelf (COTS) equipment.

DCCP: Recently, the Datagram Control Protocol working group has been chartered to develop and standardise the Datagram Congestion Control Protocol (DCCP). DCCP is a minimal general-purpose transport-layer protocol providing only two core functions:

- The establishment, maintenance and tear-down of an unreliable packet flow.
- Congestion control of that packet flow.

DCCP aims to minimise the overhead of packet header size or end-node processing as much as possible. Therefore, DCCP is as simple as possible, and as far as reasonably possible, it should avoid providing higher-level transport functionality. DCCP will provide a congestion-controlled, unreliable packet stream, without TCP's reliability or in-order delivery semantics. Additional unicast, flow-based application functionality can be layered over DCCP.

10.2.2.2 Standard Applications

Different classes of applications have very different network requirements. There are at least five classes of standard applications, having different network requirements specified in the following sections.

10.2.2.2.1 Data transfers

This class typically includes file transfer/text applications/protocols such as FTP and Email. The File Transfer Protocol (FTP) application allows file transfers between a client and a server using TCP. Email packages are sent and received using TCP. Modern Email packages use a combination of SMTP and POP (Post Office Protocol). Both SMTP and POP use TCP.

This class of applications tend to have zero tolerances for application-level packet loss but high tolerances for delay and jitter. Typical acceptable response times range from a few seconds for file transfers to minutes/hours for Emails. Bandwidth requirements in the order of Kbps are acceptable, depending on the file size, keeping the response time in order of a few seconds.

10.2.2.2.2 Video/Audio/Data Streaming

A variety of data applications such as encoded music, compressed video, and audio programming can be used over the Internet. There are two modes of transmission of produced and stored video, audio, or data over the Internet, namely the download mode and the streaming mode.

In the download mode the user downloads the entire stored file and then plays back the video or audio file. Full file transfer in the download mode usually suffers long transfer time.

Streaming refers to real-time transmission of stored video, audio, or data. In the streaming mode, the stored content is being played out while parts of the content are being received and encoded. The streaming class of applications tend to have some tolerances for packet loss, and medium tolerances for delay and jitter. Typical acceptable response times are in order of a few seconds. This is because the server can buffer multimedia data on the client to a certain degree. This buffer will then drain at a constant rate on the client side, while simultaneously receiving bursty streaming data from the server with variations of delay. As long as the buffer can absorb all variations by not draining empty, the client will see a constant video and voice. Typical bandwidth requirements are in order of Mbps for video, depending on the frame rate, compression/decompression algorithms, and size of images, or a few tens of kbps for audio streaming. The data is transmitted over UDP.

10.2.2.2.3 Interactive Video/Audio

This class of applications includes video/audio conferencing. The video conferencing application lets users transfer video across the network. So in a conference of several participants, each participant will be pumping out their voice and video streams and will simultaneously receive other participants' streams. UDP is the default transport protocol used. A voice application enables two clients to establish a virtual channel over which they can communicate using digitally encoded voice signals. UDP is the default transport protocol used for this application. Video/Audio conferencing typically is multi-point to multi-point conference calls. This class of applications tends to have some tolerances for packet loss, and low tolerances for delay and jitter due to the interactive nature of the data being transferred. Typical bandwidth requirements vary, depending on number of simultaneous participants in the conference. Response time requirements normally range from 250ms to 500msec; this is compounded by bandwidth requirements. The bandwidth requirements for each stream can be in order of Mbps.

RTP (Real Time Protocol) is the protocol used in order to provide end-to-end delivery services for real-time data, such as interactive audio and video. These services include payload type identification, sequence numbering, time stamping and delivery monitoring. RTP is run on top of UDP to make use of its multiplexing and checksum services.

10.2.2.2.4 Mission-Critical Applications

Enterprise mission-critical applications include software packages such as SAP (Systems, Applications, and Products in Data Processing), the leading ERP (Enterprise Resource Planning) software solution. ERP is a business software solution that an enterprise uses to operate its day-to-day business. It is usually comprised of several modules such as financial module, production module, distribution module, etc. Each of these modules shares information and they are totally integrated using only one database to ensure no duplication of data. SAP was the first and, to date, the most successful solution to integrate nearly all business processes such as "accounting, sales, distribution, manufacturing, planning, purchasing, human resources, analysis and other transactions" into one application for use in any business anywhere in the world. SAP applications provide an environment where transactions are synchronised throughout the entire system. SAP Company also introduced its newest major product upgrade named *mySAP.com* emphasising the shift to an e-business focus. *mySAP.com* is a fully Internet enabled architecture. Although SAP is recognised as the ERP market leader, there are competitors such as: Oracle, PeopleSoft, JD Edwards, and a range of mid-market ERP vendors who all provide similar packaged ERP applications.

This class of applications tends to have zero tolerance for application-level packet loss. Bandwidth requirements are in the order of Kbps, depending on the application. Response times vary from 500 msec to a few seconds.

10.2.2.2.5 Web-based Applications

A web page can potentially consist of many elements (or files). HyperText Transfer Protocol (HTTP) v1.0 used many TCP connections in parallel to download these elements. HTTP v1.1 uses a single TCP connection; it can download several elements sequentially or it can use a scheme called

"pipelining" to download many elements in parallel in order to make optimum use of the available bandwidth. HTML 2.0 was developed under the IETF to codify common practice. HTML 3.0 proposed much richer versions of HTML. Despite never receiving consensus in standards discussions, these drafts led to the adoption of a range of new features. The efforts of the World Wide Web Consortium's HTML Working Group to codify common practice in 1996 resulted in HTML 3.2. HTML v4.0 extends HTML with mechanisms for style sheets, scripting, frames, embedding objects, improved support for right to left and mixed direction text, richer tables, and enhancements to forms, offering improved accessibility.

This class of applications is used for web browsing and tends to have low bandwidth requirements unless large image files are associated with the request web page. The response time requirement ranges from 500msec to a few seconds.

10.2.2.3 Business Type Services & Applications

VPN Services: Virtual Private Networks (VPNs) are logically partitioned, private networks constructed over a shared or public infrastructure that utilises a range of technologies to ensure traffic separation and privacy of data, either self-implemented or provided by a service provider. A VPN can be built on the Internet or on a service provider's infrastructure. The two basic types of VPN services are Access VPNs and site-to-site VPNs.

Access VPNs provide remote access and connect telecommuters and mobile users to the corporate network over dialup, ISDN, wireless, and cable technologies. Site-to-site VPNs connect dispersed networks belonging to a single/multiple entities in order to offer e.g., Intranet and extranet connectivity, and can be used for delivering converged voice, video, and data over IP.

Currently, Cisco solutions provides site-to-site IPSec VPNs that can be deployed to connect customer remote offices to enterprise networks via IPSec tunnels, either over the Internet or over the service provider's core network. IPSec provides enhanced security features such as stronger encryption algorithms and more comprehensive authentication. IPSec has two encryption modes: tunnel and transport. Tunnel mode encrypts the header and the payload of each packet while transport mode only encrypts the payload. Cisco solutions also provide site-to-site MPLS VPNs offering secure data, voice, and video communication and QoS guarantees between corporate locations.

Application Service Provider (ASP): An ASP is an organisation that provides a contractual service to deploy, host, and manage applications for customers remotely. The ASP provides multiple customers with access to standardised applications that are owned by the ASP and deployed from a centrally managed hosting facility, typically a data centre. These applications are delivered to customers over the Internet or private networks. An ASP can offer services to a large number of customers by using a point to multi-point service model. ASPs enable their customers to deliver applications to their global workforces by providing “anytime, anywhere” access to applications remotely over the network. Applications include web-based programs, as well as windows-based, UNIX, and legacy mainframe programs that are centrally executed and maintained by the ASP.

Content distribution: Content distribution offloads work from origin servers by serving some or all of the contents of Web pages. A Content Distribution Network (CDN) consists of a collection of (non-origin) servers that attempt to offload work from origin servers by delivering content on their behalf. The servers belonging to a CDN may be located at the same site as the origin server, or at different locations around the network, with some or all of the origin server’s content cached or replicated amongst the CDN servers. For each request, the CDN attempts to locate a CDN server close to the client to server the request, where the notion of “close” could include geographical, topological, or latency considerations. With content distribution, the origin servers have control over the content and can make separate arrangements with servers that distribute content on their behalf.

Voice over IP trunks: This is for telecom companies/ISPs that need point-to-point bulk transfer of VoIP calls with QoS guarantees across network/s.

Video Conferencing: This is to provide high-speed videoconferencing for a large number of users such as educational institutes and research bodies.

Video on demand: This is to provide high-speed digital video for a large number of users.

Database: A database application enables the user/s to store information. Database operations are divided into two categories: a database entry and a database query. A database entry results in a fixed amount of data being written into the database. A database query results in the client issuing a query, and the server responding with some data. The default transport protocol for the database application is TCP.

Collaborative computing: The field of collaborative computing encompasses the application of computers for coordination and cooperation of two or more people who attempt to perform a task or solve a problem together. Creation of shared workspaces among collaborators is required. Video/audio conferencing is simply one form of collaborative computing where shared computer-based applications (e.g., shared editors, whiteboards) are supported in real-time. Collaborative computing sits at the crossroads of many different disciplines: multimedia, distributed systems, networking, and human factors, and so on.

Other applications: The range of network applications is wide and tedious to name all. As examples of other applications, it is possible for a manufacturer of a large and complex product to share simulations, computer renderings, and designs with hundreds of people at different locations simultaneously. Other examples also allow surgeons working at a lab in one city to use 3-D body scanning and robotics to manage a medical operation occurring elsewhere in the country.

10.2.2.3.1 Network handling of applications

The Cisco view on class of service mapping for applications in a DiffServ environment is shown in the following table.

Class of Service	Application examples
Premium-Class	VoIP, multicast Share price quotes, etc.
Business-Class	SAP, Oracle, Citrix[1], etc.
Best-Effort	Database access and replication, backups, etc.

Table 6 Class of Service Mapping to Applications

10.2.3 Traffic

Cisco states that more than 90% of sessions transfer ten packets or less each way. This is used by transaction mode (mail, small web page). More than 80% of all TCP traffic results from less than 10% of the sessions, in high rate bursts.

In Abilene network (the Internet2 backbone), carrying mostly university-to-university traffic, the percentage of `mail' traffic (which includes ports for protocols such as SMTP, POP3, IMAP, and encrypted versions of these, etc., all combined) comprises about 0.4% of traffic volume in octets and 0.6% of packets.

[1] Citrix provides a solution for centralisation and consolidation of traditionally distributed client applications and their associated data. In a server-based computing environment, the applications are installed and executed on Citrix MetaFrame servers rather than distributed client PCs. This centralisation and consolidation drives the need for enterprise networked storage with shared, multi-platform access, optimised delivery, point-in-time recovery, remote data replication, and high speed backups. Citrix applications and portal server solutions enable organisations to leverage the Internet and deliver mission-critical applications.

10.3 Intra-domain Traffic Engineering

10.3.1 IP Traffic Engineering proposals

In this section, we list some proposals that aim to add traffic engineering functionality to IGPs (Interior Gateway Protocols) such as OSPF (Open Shortest Path First, [RFC2328]) and IS-IS.

10.3.1.1 Optimising OSPF Weights

[Fortz00] studies the problem of setting weights in order to perform OSPF routing. One of the requirements addressed by [Fortz00] is to have an OSPF behaviour near that which could be obtained by deploying MPLS (Multi- Protocol Label Switching) technology in terms of flexibility to give good load balancing. From this standpoint, [Fortz00] proposes the introduction of a local search heuristic that uses hash tables to avoid cycling and for search diversification.

For more mathematical details, refer to [Fortz00].

10.3.1.2 A Flow-Based Approach

The [Rie01] proposal tries to solve the problem of the choice of the metric's value in order to optimise the routing problem. Specifically, it presents an optimisation model for IP networks, which deploys conventional shortest-path next-hop routing protocols. [Rie01] shows how metric values have to be set in order to achieve advantageous path constellations. In addition, this proposal is not restricted to purely additive link metrics.

Within the context of this proposal, and assuming the two metric types bandwidth and delay, different variants of the traffic-engineering problem are examined:

- Both metrics can be chosen independently of the installed hardware, thus, allowing the highest degree of freedom for optimisation purposes.
- One metric type is set for the given network infrastructure (e.g., fixed *bandwidth* components in an EIGRP network), and the other metric value serves as an optimisation variable.
- One metric type is ignored, and route optimisation is based solely on the other metric.

10.3.1.3 Traffic Engineering Extensions to OSPFv2

[Kat01] describes extensions to the OSPFv2 [RFC2328] protocol to support intra-area Traffic Engineering using Opaque Link State Advertisements (LSA Opaque). These extensions provide a way of describing the traffic engineering topology and distributing this information within a given OSPF area.

Within the context of this document, two top-level TLVs are defined:

- Router Address
- Link

In addition, the following sub-TLVs are defined:

- Link type
- Link ID
- Local interface IP address
- Remote interface IP address
- Traffic engineering metric
- Maximum bandwidth

- Maximum reservable bandwidth
- Unreserved bandwidth
- Administrative group.

The information carried in these LSA Opaque is flooded in the network and used to build the traffic engineering routing database.

Note that those extensions are applicable for intra-area distribution of Traffic Engineering information. Methods of inter-area and inter-AS are not discussed within the context of [Kat01]. For more details refer to [Kat01].

10.3.1.4 Traffic Engineering Extensions to OSPFv3

[Ish01] describes extensions to the OSPF version 3 [RFC2740] to support intra-area Traffic Engineering and expands the extensions proposed in the [Kat01] in order to make it applicable both IPv4 and IPv6 network. New sub-TLVs are defined to support IPv6 network. These new sub-TLVs are not limited to use in OSPF version 3 but can also be used in OSPF version 2.The Three new sub-TLVs proposed in [Ish01] are:

- Neighbour ID
- Local Interface IPv6 address
- Remote Interface IPv6 address.

10.3.2 MPLS Intra-domain Traffic Engineering

The problem of traffic engineering has attracted a lot of attention in recent years. Traffic Engineering entails the aspect of network engineering that is concerned with the design, provisioning, and tuning of operational Internet networks. In order to deal with this important emerging area, the Internet Engineering Task Force (IETF) has chartered a Traffic Engineering Working Group (TE-WG) to define, develop, specify, and recommend principles, techniques and mechanisms for traffic engineering in the Internet. The main output of this working group until now is that it has defined the basic principles for traffic engineering [RFC3272] and the requirements to support the interoperation of MPLS and DiffServ for traffic engineering [Fauch02]. It is in the plans of this group to look into technical solutions for meeting the requirements for DiffServ-aware MPLS traffic engineering, the necessary protocol extensions, interoperability proposals and measurement requirements.

Two similar works with the work presented in this chapter are Netscope [Feld01] and RATES [Aukia00]. Both of them try to automate the configuration of the network in order to maximise network utilization. The first one uses measurements to derive the traffic demands and then by employing the offline algorithm described in [Fortz00] it tries to offload overloaded links. The latter uses the semi-online algorithm described in [Kodi00] to find the critical links which if they are chosen for routing will cause the greatest interference (i.e. reduce the maximum flow) of the other egress-ingress pairs of the network. Both of these works do not take into account any QoS requirements and only try to minimize the maximum load of certain links.

The algorithm described later in this chapter can be categorised as (class-based) *offline traffic engineering* [RFC3272]. Such problems can be modelled as multi-commodity network flow optimisation problems [Ahuja93]. The related works use optimisation formulations, focusing on the use of linear cost functions, usually the sum of bandwidth requirements, and in most of the cases they try to optimise a single criterion, minimize the total network cost, or combine multiple criteria in a linear formula.

In Mitra et al [Mitra99a] the traffic-engineering problem is seen as a multi-priority problem which is formulated as a multi criterion optimisation problem on a predefined traffic matrix. This approach uses the notion of predefined admissible routes which are specific for each QoS class and each source-destination pair, where the objective is the maximization of the carried bandwidth. In [Mitra99b], the

authors address the resource allocation and routing problem in the design of Virtual Private Networks (VPNs). The main objective is to design VPNs which will have allocated bandwidth on the links of the infrastructure network such that, when the traffic of a customer is optimally routed, a weighted aggregate measure over the service provider's infrastructure is maximized, subject to the constraint that each VPN carries a specified minimum. The weighted measure is the network revenue, which is a function of the traffic intensity. The algorithm proposed in that paper solves first the optimal routing problem for each VPN independently. Then it calculates for each VPN the linear capacity costs for all the links. These quantities are used to modify appropriately the current capacity allocations so that the network revenue of the infrastructure network for the new capacities is maximized. In [Mitra99b], it is shown that this is equivalent to minimizing a linear function of the capacity costs subject to constraints imposed by the link capacities.

In [Poppe00] a model is proposed for off-line centralized traffic engineering over MPLS. This uses the following objectives: resource-oriented or traffic-oriented traffic engineering [RFC3272]. The resource-oriented problem targets load balancing and minimization of resource usage. Capacity usage is defined as the total amount of capacity used and load balancing is defined as one minus the maximal link utilization. The objective function that has to be maximized is a linear combination of capacity usage and load balancing, subject to constraints imposed by the capacity of the links. The traffic-oriented model suggests an objective function that is a linear combination of fairness and throughput, where throughput is defined as the total bandwidth guaranteed by the network and fairness as the minimum weighted capacity allocated to a traffic trunk. In [Suri01], the authors propose an algorithm which has two phases, a pre-processing phase and an on-line one. In the pre-processing phase the algorithm uses the notion of multi-commodity flows, where commodities correspond to traffic classes. The goal is to find paths in the network to accommodate as much traffic as possible from the source to the destination node. The algorithm tries to minimize a linear cost function of the bandwidth assigned to each link for a traffic class. The second phase performs the on-line path selection for LSP requests by using the pre-computed output of the multi-commodity pre-processing phase.

Works like [Fortz01] [Wang01] [Breit02] try to achieve optimal routing behaviour by appropriately configuring the shortest path routing metrics, assuming no MPLS is supported. Wang et al. in [Wang01] proved theoretically that any routing configuration, including the optimal one, could be achieved by the appropriate setting of the shortest path routing metrics.

Finally, regarding online algorithms, they are mainly based on extensions of the QoS-routing paradigm [Chen98]. These approaches are heuristics, recently known in the IETF as Constraint-Shortest Path First (CSPF), which utilise information kept in traffic engineering databases populated through information obtained from the routing flooding mechanisms about link capacities, unreserved capacity, colour affinities etc. Other online traffic engineering approaches [Elwal01], [Cao00] mainly focus on load balancing on multiple equal or non-equal cost paths

10.4 Inter-domain Traffic Engineering

10.4.1 Introduction

This chapter portrays the state of the art on inter-domain traffic engineering. Techniques for inter-domain traffic engineering are especially important to MESCAL, since any QoS enabled traffic delivery spanning more than one domain, requires management on the links between ASes. This chapter covers applicable BGP developments as well as the state of the art on the emerging inter-domain extensions to Multi Protocol Label Switching (MPLS).

In order to enable an inter-domain QoS path, it may be beneficial or necessary for the exterior gateway protocol to carry QoS related information. Detailed in this chapter are several proposals, the ability to define more than one route per destination, the distribution of QoS information per advertisement and the definition of TE weights on routes and route bandwidth advertisement capabilities.

The emerging trends on inter-domain MPLS could allow end-to-end LSPs to be established between users connected to different ASes. MPLS is not explicitly part of any MESCAL inter-domain solution,

as it is not a layer-3 protocol. However, it should be considered as an alternative solution. The proposals discussed include the recent internet drafts on realising inter-domain end-to-end LS path set-up, an RFC on BGPv4 label distribution, RSVP extensions to enable inter-domain LSP establishment and a study on the cost of MPLS or partial MPLS inter-domain solutions. Finally, both CISCO and Juniper have recently provided functionality for inter-domain LSP handling in their router operating software.

10.4.2 BGP

In this section, we focus on proposals to use BGP-4 as an inter-domain routing protocol in order to carry QoS (Quality of Service) information. The BGP-4 specification [RFC1771] provides some attributes that can be used to implement routing policies. Whatever the solution, BGP nodes should implement the BGP capabilities advertisement [RFC2842] in order for domains to agree on what they can do as far as QoS is concerned.

In this section, the proposals are presented briefly, for more details refer to the related documents.

10.4.2.1 Using BGP policies

The BGP-4 protocol, as defined in [RFC1771], provides some attributes than can be used to discriminate among several routes towards the same destination, especially the LOCAL_PREF and the MED attributes:

- LOCAL_PREF: the local preference attribute is used within a domain to associate a degree of preference to each exit point to join a specific destination. This attribute is meaningful only within a domain and is not forwarded to external peers.
- MULTI_EXIT_DISC: the multi exit discriminator is an attribute that is meaningful between two neighbouring domains that are connected via several external BGP peers. It is used to choose the best exit/entry point between these two domains.

In addition, the COMMUNITIES attribute is defined in [RFC1997]. It consists of a set of "communities", each "community" being coded with a 4 octets value, and aims at exchanging additional information to neighbouring and remote BGP domains. It is used to control routing information distribution by grouping IP prefixes according to the communities they belong to. Therefore, a common policy can be applied to all prefixes of the same community.

This attribute is optional and transitive. This means that a domain that does not implement such an attribute must forward the COMMUNITIES attribute to its peers, even though it does not take into account its value within its own domain.

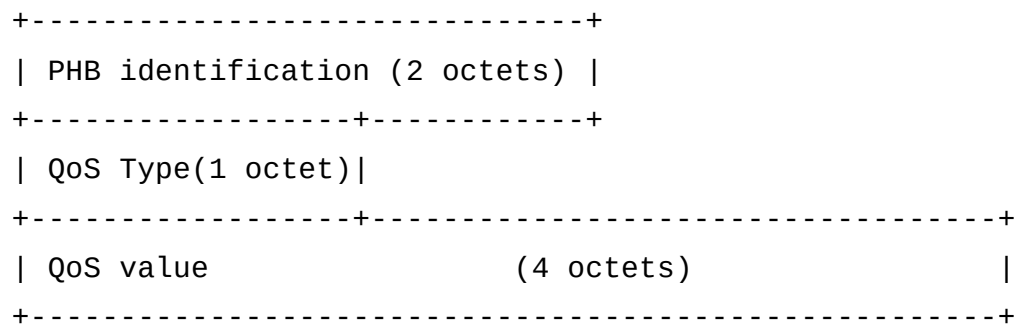
10.4.2.2 Using BGP to distribute flexible QoS information

The draft [Bon01] proposes a flexible QoS attribute that can be used to distribute QoS information with BGP. The flexibility of the proposed attribute allows each AS to decide independently which QoS information to redistribute to its peers.

The proposed attribute allows association of a set of supported PHB, transit delay and bandwidth information to an UPDATE message. The QoS attribute is a variable length non-transitive optional attribute.

- The attribute flags shall indicate that the QoS attribute is optional, non-transitive and the extended length bit is set to one since the QoS attribute may be longer than 256 bytes.
- The attribute type code is to be assigned by IANA.
- The length of the entire attribute is encoded in two octets.

The value of the QoS attribute is encoded as a list of triples:



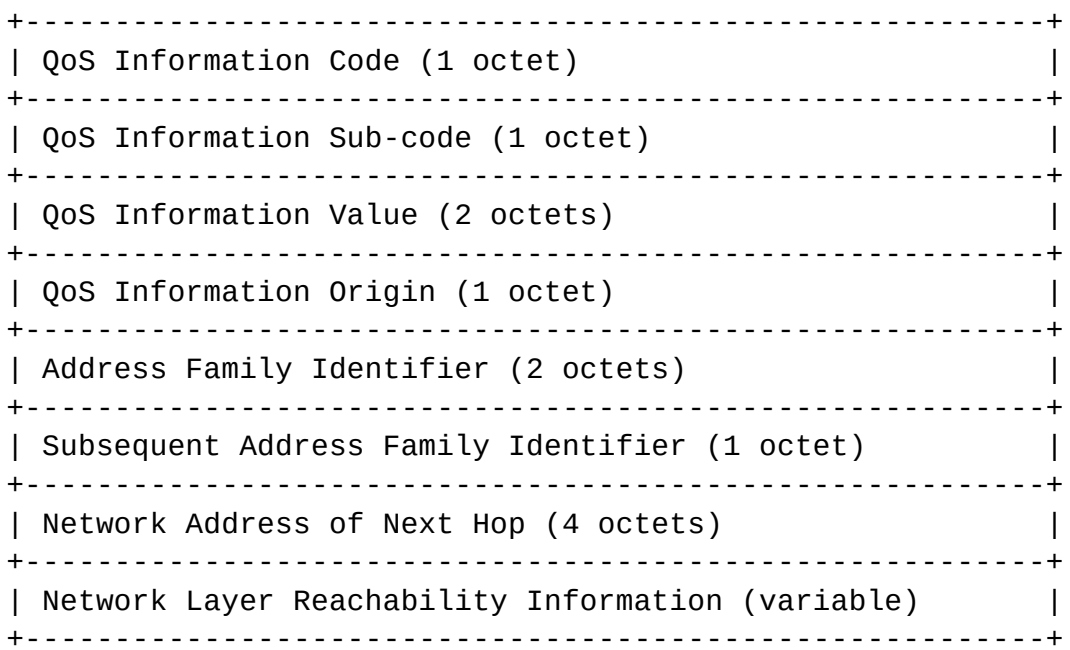
The QoS type code allows the definition of 256 different types of QoS values. The value 0 is reserved for future utilisation; values 1-127 are to be defined by IANA while values 128-255 are reserved for vendor specific QoS attributes.

The QoS types defined within the context [Bon01] are:

- Empty QoS value: used by the BRs (*Border Routers*) to announce the support of a specific PHB towards the associated prefix with associating detailed QoS information.
- Maximum Bandwidth: used by the BRs to associate a maximum bandwidth with a given PHB.
- Available Bandwidth: used by the BRs to announce the available bandwidth associated with an announced prefix.
- Maximum Transit delay: used by the BRs to associate a maximum transit delay with an announced prefix.
- Minimum Transit delay: used by the BRs to associate a minimum transit delay with an announced prefix.

10.4.2.3 Providing QoS indication with the BGP-4 protocol: the QoS_NLRI attribute

One proposal to exchange QoS attributes between domains is specified in [Cri01]. This defines a new attribute (QOS_NLRI) for BGP-4. This attribute aims at associating QoS information with IP prefixes. The QOS_NLRI attribute is defined as follows:



The meanings of the fields are as follows:

- QoS Information Code: type of QoS information (packet rate, delay, jitter, PHB identifier).
- QoS Information Sub-code: sub-type of QoS information (reserved rate, available rate, loss rate, min/max/average delay).
- QoS Information Value: value of the QoS information identified in the 2 previous fields.
- QoS Information Origin: Origin of the path information as described in BGP-4 specification [RFC1771].
- Address Family Identifier: network layer protocol associated with NLRI prefixes.
- Subsequent Address Family Identifier: additional information about NLRI prefixes.
- Network Address of Next Hop: IPv4 next-hop address.
- Network Layer Reachability Information: prefixes associated with QoS information.

This attribute is optional and transitive. This means that a domain that does not implement such an attribute must forward the QoS information to its peers, even though it does not take into account this information within its own domain.

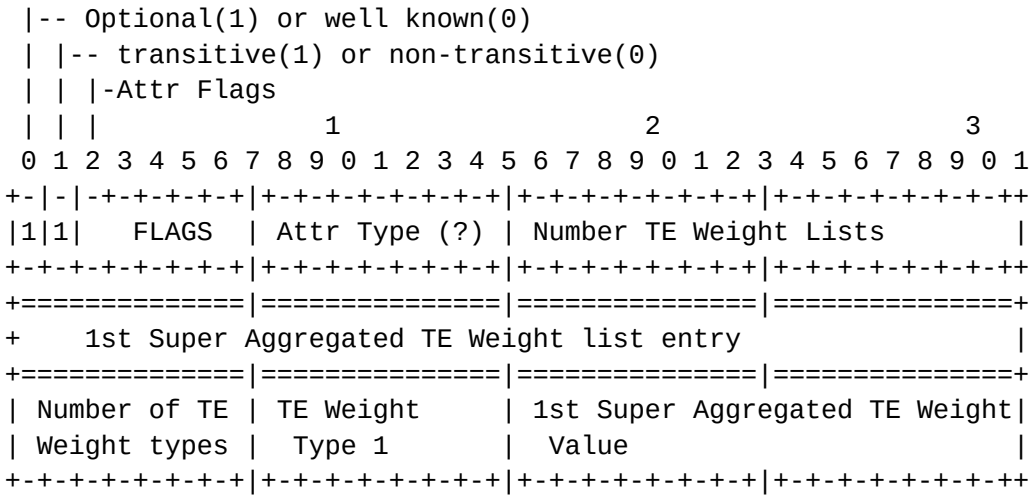
10.4.2.4 Use of BGP TE weights

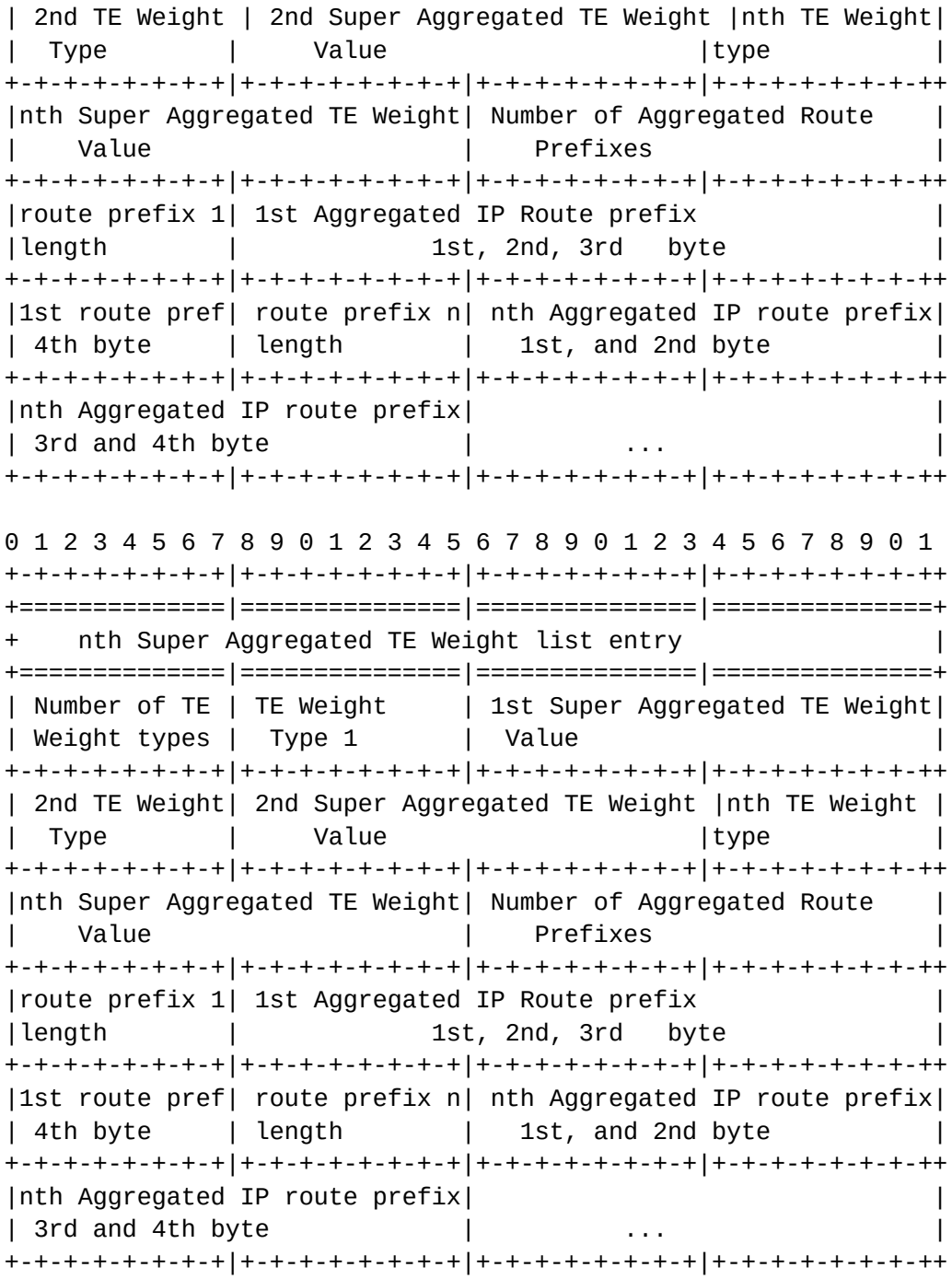
The main idea of the [Aba01] proposal is that the BGP protocol can be utilised to choose the best BGP routes based on traffic engineered constraint weights. This information can be propagated between all BGP peers and calculated by the BGP AS border routers before it is deployed to their forwarding tables.

In order to propagate constraint summarisation weights for each AS, [Aba01] propose to define a new attribute, which contains QoS information called TE weights that can include some QoS parameters (e.g. bandwidth, number of hops, delay and QoS service classes). Each BGP router would propagate this information to its peers.

When the BGP RIB database is loaded with TE Weight information, a TE capable BGP router would compute based on TE manual configuration criteria the best BGP route for a given destination. The BGP Route Selection process is extended to support a TE way of prioritising the best routes for any configured destination, in which the order and preference of the routes can be changed to give the TE weight attribute a higher priority than other attributes.

The TE weight attribute format is as follow:





For additional details refers to [Aba01].

10.4.2.5 BGP and the use of Available Bandwidth Index

[Xia01] introduces a new QoS metric called ABI (Available Bandwidth Index) to perform the bandwidth advertising and routing. The ABI metric is defined as an association of an interval and a probability (For more mathematics details refer to [Xia01]). Authors of [Xia01] justify the introduction of such metric by the stability of this value in time scale. From this standpoint, three modifications to the BGP protocol are to be considered:

- Extend BGP UPDATE messages to record QoS information.
- Select paths based on the QoS information stored in the extended BGP UPDATE messages.
- Monitor and update the QoS state of the advertised routes.

10.4.3 MPLS-based Inter-domain TE

10.4.3.1 Introduction

Currently, all inter-domain connectivity on the Internet is achieved with BGP prefix advertisements. If a prefix is advertised by an AS, traffic can be routed to that destination prefix through the AS. There is no traffic conditioning at the inter-domain border and no control over data throughput. No guarantees can therefore be made on traffic delivery. MPLS label switched paths can be engineered to deliver traffic with hard QoS guarantees, however, MPLS is currently limited to intra-domain use. In order to extend the reach of LSPs beyond the borders of a domain, it is necessary to first advertise available LSPs via an external gateway protocol such as BGP and then to use admission control at the border so as to control the quantity of paths that are set up. Such implementation does not imply that conventional IP traffic will be replaced by MPLS LSPs but that these can be advertised in addition to the currently available best-effort traffic.

The MESCAL project does not specify the use of MPLS for any part of the architecture. It also explicitly states the use of layer-3 technology – DiffServ and BGP in particular – for all inter-domain signalling requirements. Several of the Internet Drafts discussed in this chapter are neither DiffServ nor QoS aware, although they may be extended in order to comply with MESCAL requirements.

All proposals referenced in this document are summarised, for more information please refer to the original document.

10.4.3.2 Cisco Internet Drafts

10.4.3.2.1 MPLS TE Requirements

An Internet Draft discusses Service Providers' requirements for the support of inter-AS MPLS Traffic Engineering (MPLS TE) [Zhang03]. The document presents a set of requirements and deployment cases in order to arrive at some general guidelines and definitions. The requirements build on two scenarios:

- A service provider network made of multiple ASes, requiring "intra-SP" TE to form a "seamless TE plane."
- An "inter-SP" scenario based on TE peering agreements: this expands to a number of sub-scenarios
  - MPLS tunnels between two ASs.
  - tunnels to interconnect ASs of one SP through the network of another SP(s).
  - inter-AS tunnels linking customer end points.

The issues in current BGP-based traffic engineering practices across multiple ASes are discussed before arriving at the requirements for extending the current MPLS TE mechanisms beyond AS boundaries. The paper covers inter-AS TE in IPv4 and VPNv4 addressing planes. Identified requirements are in areas of operations and interoperability, management, DS-TE support, optimality, support for diverse paths, fast rerouting, signalling and path computation, aggregation/hierarchy, mapping of traffic, re-optimisation, confidentiality and policy control.

Applicability to MESCAL: Manifestations of the requirements may vary widely for the two scenarios, as some business sensitive information may not be available to "inter-SP" peering, leading to a less flexible solution. While the MESCAL project is mainly interested in the inter-SP scenario, it may also find viable solutions for the intra-SP case. An important point is the lack of requirements for QoS awareness and path optimality based on anything other than shortest path or load balancing. These are required for MESCAL and would need to be addressed.

10.4.3.2.2 MPLS TE Solutions

Building on the requirements established in [Zhang03], an Internet Draft proposed by J.P.Vasseur [Vasseur03] formulates a set of mechanisms to establish and maintain MPLS Traffic Engineering Label Switched Paths spanning multiple Autonomous Systems. A number of drafts specify various subsystems that are required for these mechanisms [Vasseur02], [Vasseur01], [Vasseur00]. The solution draft specifies two modes of operation, a distributed computation and a centralised computation, both being carried out separately inside each AS.

The distributed mode is realised by means of loose LS path routing. Every inter-AS LSP path in this mode is defined as a set of loose and strict hops, but at a minimum the AS border routers (ASBR) traversed by the LSP must be specified. This requires each ASBR on the route to be pre-computed and statically assigned to the LSP at the head end. The draft introduces an inter-domain flooding mechanism to provide this information, however, it is uncertain if a BGP extension could provide for this requirement. Each router whose next hop is loose computes the route to this next loose hop (ERO expand) meeting a given set of constraints. The route taken by the LSP is hence optimal only from the viewpoint of each loose hop router.

The second mode of operation specifies centralised path computation by means of one or more path computation servers (PCS), which must maintain the full topology of the AS they are serving. Depending on the complexity of the PCS computations, the server may be co-located with ASBRs or a standalone server inside the AS. The path computation server needs to be statically or dynamically locatable before path computations can be requested. Path computation requests are relayed from one path computation server to the next (backward recursive computation) until the whole end-to-end route is resolved. The full path is not laid open to any single PCS, thereby protecting private routing information. The draft [Vasseur02] describes the signalling for requesting path computations.

Applicability to MESCAL: At present each PCS is defined to compute a shortest path route for each request, but for the case of multiple AS paths to a particular destination or even multiple ASBRs to the same destination, this may be a complex problem to solve without sufficient knowledge of the overall path. This particular problem may be of special interest to the MESCAL project, as this presents a means to induce inter-domain TE information into ASes that is otherwise inaccessible. MESCAL could extend the shortest path computation, providing for QoS and other constraint based routing.

The distributed route computation model also has no mechanisms for QoS or constraint-based routing in its current state. In order for MESCAL solutions to implement the distributed model, some extensions may be required to provide access to the route computation components of each path. These could be any LSRs inside the AS or just the ASBRs, but in the latter case the distributed scenario converges to a version of the centralised scenario.

It may also be important to notice that the draft merely specifies *mechanisms* for inter-domain MPLS. There are no descriptions to solve technical problems such as scalability issues or how information about ASBRs is passed between ASes. Methods for passing ASBR or PCS locations could potentially suffer scalability limitations, since multiple route information needs to be passed for each destination. The PCSs need to be made aware of potentially thousands of AS neighbours, with the problem getting worse when utilising the distributed computation approach.

In addition to [Vasseur03], there are two drafts [Vasseur01], [Vasseur00] that enable the re-optimisation of LS paths and a fast reroute for link failure protection. The re-optimisation mechanism allows a Head-End LSR, or any other router on the path to initiate a re-computation of the loosely routed path, in order to ensure that a better path – if available – can be utilised. The draft appears to target long-lived LS paths, such as could arise from the tunnelling scenarios in [Vasseur03]. The fast reroute draft proposes to specify an additional flag of the RRO IPv4/IPv6 sub-object used for the backup tunnel selection for inter-area/inter-AS TE LSP protected by MPLS TE Fast Reroute in case of ABR/ASBR node failure.

Applicability to MESCAL: An implementation of the route optimisation draft may be of use to some MESCAL solution scenarios, although the emphasis here is on end-to-end QoS with shorter lifetime rather than static inter-AS paths. There is no explicit requirement for resilience or route optimisation defined within MESCAL. However, if it becomes necessary to implement such functionality, these drafts should be considered.

10.4.3.3 Other Research

An RFC is available on the distribution of label information with BGP-4 [RFC3107]. It specifies the way in which the label mapping information for a particular route is piggybacked in the same BGP Update message that is used to distribute the route itself. When BGP is used to distribute a particular route, it can be also be used to distribute an MPLS label, which is mapped to that route. The RFC also specifies multiple labels for a prefix and the possibility of advertising more than one route per prefix, provided that these routes have different labels assigned.

The problem of establishing explicitly routed inter-domain LSPs is discussed in an Internet Draft [Pelsser02], showing that the current sub-objects found in RSVP-TE are not sufficient to establish inter-domain LSPs, because they do not take into account the policy constraints of the inter-domain environment. The draft looks at the possibility of protecting segments of inter-domain LSPs. It also describes the necessary RSVP objects and flags and discusses the impact of the solution on the syntax of existing RSVP-TE objects and the syntax of new required objects that are presented.

Applicability to MESCAL: A means of label distribution may prove a viable option to MESCAL should MPLS become a means of inter-domain transport.

A paper studied the cost of using MPLS for inter-domain traffic for both MPLS and hybrid MPLS and IP cases [Uhlig00].

10.4.3.4 Implementation

The Juniper JUNOS Internet software provides traffic engineering tools to configure MPLS so as to control the paths that traffic takes to destinations outside an AS [Juniper]. Both IBGP and EBGP take advantage of the LSP host routes. BGP compares the BGP next-hop address with the LSP host route. If a match is found, the packets for the BGP route are label-switched over the LSP. If multiple BGP routes share the same next-hop address, all the BGP routes are mapped to the same LSP route, regardless of which BGP peer the routes are learned from. If the BGP next-hop address does not match an LSP host route, BGP routes continue to be forwarded based on the IGP routes within the routing domain. In general, when both an LSP route and an IGP route exist for the same BGP next-hop address, the one with the highest preference is chosen.

Cisco IOS software contains an implementation of RFC 3107 (discussed above) as of version 12.0(21)ST [Cisco]. The functionality allows setting up a VPN service provider network so that the autonomous system boundary routers (ASBRs) exchange IPv4 routes with MPLS labels of the provider edge (PE) routers.

10.5 Signalling Protocols

10.5.1 BGRP

Border Gateway Reservation Protocol (BGRP) [BGRP] is a signalling protocol for inter-domain aggregated resource reservation for unicast traffic. BGRP builds a sink tree for each of the stub domains. Each sink tree aggregates bandwidth reservations from all data sources in the network. BGRP maintains these aggregated reservations using soft state and relies on Differentiated Services for bandwidth reservation.

The same authors in [BGRP] submitted an Internet draft [BGRP-fm] in January 2000. This draft first defines the scaling problem in today's Internet backbone, and briefly discusses several existing resource management approaches. Then, it presents a distributed approach and introduces the BGRP,

for inter-domain resource reservation that can scale in terms of message processing load, state storage and control message bandwidth. The main idea of this approach is to build a sink tree for each domain network. Each sink tree aggregates reservations from all data sources in the network. Sink tree initiation, maintenance, and termination involve only backbone border routers. Within each domain, the network service providers manage network resource and direct user traffic independently. At the border routers, the service providers can use BGRP to set-up domain-level reservation trunks base on bi-lateral agreement. Since routers only maintain the sink tree information, the total number of reservation states at each router scales, in the worst case, linearly with the number of domains in the Internet. For bandwidth reservation, BGRP relies on DiffServ. As a result, the number of packet classifier entries is small. To reduce the protocol message traffic, routers may reserve domain bandwidth beyond the current load so that sources can join or leave the tree or change their reservation without having to send messages all the way to the root for every such change. This Internet draft expired in July 2000, meaning that its original authors either didn't find the time to update the document and to pursue the work or decided to abandon the work .

Recently, two related Internet drafts are submitted by Aquila consortium [BGRP+] and [BGRP-per].

The BGRP+ presents an approach to inter-domain bandwidth reservation requests and allocation, based upon an enhanced utilisation of BGRP. BGRP uses a "sink tree" based aggregation for resource reservations over a network of DiffServ domains. However, aggregation of reservations is just the first step towards scalability. To limit the signalling load and the processing power required in the BGRP agents, it is also necessary to reduce the number of signalling messages. The proposed enhancements in this draft rely upon a mechanism - the "Quiet Grafting" mechanism, which allows a significant reduction of the BGRP signalling messages overhead so that not each message has to travel edge-to-edge through the DiffServ network region.

It is shown in [BGRP] that lack of quiet grafting mechanisms may lead to scalability problems, especially for short-lived small bandwidth flows. Transit domains are mainly affected by this behaviour. In [BGRP+], a first perspective to quiet grafting mechanisms is given, while in the [BGRP-per] draft these mechanisms are further analysed and verified through simulation studies. It is shown in [BGRP-per] draft that with the deployment of the quiet grafting mechanisms, the number of signalling messages is reduced, since each signalling message does not have to travel all the way from the source to the destination domain.

10.5.2 SIBBS

The Simple Inter-domain Bandwidth Broker Signalling (SIBBS) protocol is the work of the QBone Signalling Workgroup that was done in 1999-2000. The Bandwidth Broker is a software entity that resides in a DiffServ domain and manages resources for IP QoS services. It is responsible for negotiating with bandwidth brokers from neighbouring domains. It is also responsible for internal and external admission control decisions according to policies held on a database. Based on such decisions it configures any routers within its domain. The protocol consists of a simple request-response protocol between the bandwidth broker peers that carries the essential information for requesting a service in general. In general, a bandwidth broker may receive a resource allocation request (RAR) from either an element in the domain that the bandwidth broker controls (or represents), or a request from a peer (adjacent) bandwidth broker. In either case, the bandwidth broker responds to this request with a confirmation of service or denial of service. This response is known as a Resource Allocation Answer (RAA). The response may contain messages, such as altering the router configurations at the access, at the inter-domain borders, and/or internally within the domain, and possibly generating additional RAR messages requesting downstream resources. RARs flow inter-domain between peer (adjacent) bandwidth brokers. The mechanism for triggering the response is defined in the SIBBS protocol specification.

The basic assumption for SIBBS is that SLSs are already established (pairwise) between peer BBs "out-of-band", that is, without a SLS negotiation protocol. It is also assumed that there are globally well-known services (GWS) and service IDs (GWSID) referring to those services. The SLSs refer also to these services and in addition, resource allocation requests use the well-known IDs. Further, the BB

handles end system requests for its domain, and BBs may peer directly with non-adjacent BBs. The latter is to facilitate the aggregation of service requests.

Assuming statically configured SLAs and SLSs between adjacent domains, the service is then **realised** by the bandwidth broker receiving a RAR and subsequently a RAA by configuring the routers at the edges of (and internal to) its domain with the set of parameters for the PHBs and the traffic conditioning mechanisms derived from the RAR, the service definition, and the SLS in place with the peer domains.

Lastly, it is assumed that bandwidth brokers communicate with one another via long-running TCP sessions and that the reliability and flow control provided by TCP are sufficient for this application. The long-running TCP connections are established with out-of-band information; that is, the knowledge of names and IP addresses of peer bandwidth brokers is spread via some human interface or external protocol. The globally well-known service specified in the RAR messages in this protocol must be mapped by individual DS domains to DSCPs which in turn specify PHBs in the routers handling the DiffServ aggregates. This mapping is left to the individual domains. Finally, the source end system (or Bandwidth Broker) receives the RAA and is able to send the flow.

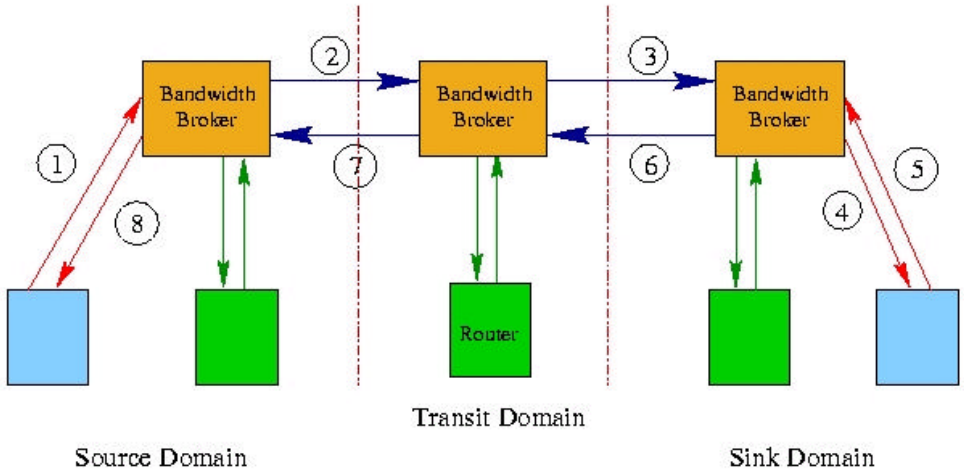
Three different cases are considered in the protocol:

- **Case 1**: A source end system initiates a request to its BB for service to a destination end-system with a fully-specified destination address.
- **Case 2**: An end system or a bandwidth broker request service (i.e., a pipe) to another domain with a destination prefix.
- **Case 3**: A bandwidth broker receives a request for service with fully-specified destination prefix but uses a pipe ("core tunnel") to satisfy the request.

The first scenario shows the basics of inter-domain bandwidth broker communication. It is not expected that the entire mechanism will be used for every request in the network. This would not be especially scalable. The variations in the other two scenarios can be used to support aggregation and increase scalability.

10.5.2.1 Case 1: Service request initiation from an end-system to another one

Figure 47 shows an overview of the communication involved in this scenario. The request proceeds hop-by-hop and is sent only between "adjacent" entities.



End system request with fully specified destination

Figure 47 End system request with fully specified destination

- *Behaviour of the bandwidth broker in the originating domain:*

  The source end system sends an RAR to its domain BB (**1**). This message includes a globally well-known service ID and an IP destination IP address, a source IP address, an authentication field, times for which the service is requested and the other parameters of the service.

  The bandwidth broker makes a number of decisions including the following:

  a. Whether the requester is authorised for this service.

  b. The egress router to which the flow exits.

  c. The internal route through the domain to the egress router.

  d. Whether the flow fits in the SLS of the egress router with the net domain in the path to the destination.

  e. Whether the flow is accepted for the specified service possibly based on the domain policies.

  If these decisions all have a positive outcome, the BB will modify the RAR by including the ID for the domain, sign the request with its own signature, and pass the RAR to the adjacent BB.

  In the case where these decisions have negative outcomes, then the bandwidth broker returns a Resource Allocation Answer (RAA) to the source end system including additional information such as a reason code for the rejection, etc (**8**).

- *Transit domain handling of the request*

  In this case, the bandwidth broker receives an RAR from an adjacent bandwidth broker with a fully-specified destination address specification (**2**). The transit bandwidth broker must perform a number of functions:

  a. Authenticate that the request is indeed from a peer bandwidth broker.

  b. Determine egress router from its (inter-domain) routing tables.

  c. Check that the requested resources fall within the SLS with the preceding domain that sent the RAR connected via one of the ingress routers of this domain.

  d. Check that the requested resources fall within the SLS connecting to a successor domain en route to the destination.

  e. Ensure that there are sufficient resources within the domain to support the flow from the ingress border router and determine the intra-domain route.

  f. Determine whether the flow is accepted for the specified service possibly based on the domain policies.

  In the event that all these decisions have positive outcomes, the transit bandwidth broker modifies the RAR as appropriate (e.g. putting its own ID in the sender's ID field and authentication string in the message) and sends it to the bandwidth broker of the following domain en route to the destination IP address **(3)**.

  In the event that these decisions have negative outcomes, the BB returns an RAA to the sending domain **(7)**.

- *Behaviour of the bandwidth broker in the destination domain.*

  In this scenario, the bandwidth broker of the destination domain knows the address of the end system which is to receive the flow. As in the behaviour just described, on the reception of the RAR **(3)**, the BB makes the following decisions:

  a. Authenticate that the request is indeed from a peer bandwidth broker.

b. Determine the intra-domain route from the ingress router to the end system and decides whether the resources are available to support the flow.

c. Check that the requested resources fall within any possible SLS with the end system.

d. Determine whether the flow may be accepted.

In case these decisions have negative outcomes, an RAA is sent back **(6)**, possibly with a reason code and hints about acceptable parameters.

In case all these decisions have positive outcomes, the bandwidth broker sends the RAR to the end system with appropriate changes **(4)**. In this case, the destination end system makes the determination whether it can receive the flow. This is signalled with an RAA to the bandwidth broker of the destination domain **(5)**. The RAA contains authentication of the destination end system, and parameters for the flow which the destination end system is willing to accept (which may be different from those received). In case the flow is rejected, the RAA contains a reason code and possibly hints about the set of service parameters that would be acceptable.

Upon receiving the RAA from the destination end system **(5)**, the bandwidth broker authenticates the answer and forwards the RAA, with appropriate changes to the peer bandwidth broker that sent the RAR **(6)**. At the same time, the bandwidth broker may configure traffic conditioners at the ingress router and possibly at other routers along the intra-domain path to the destination.

- *Transit domain processing of the RAA*

  The RAA received from the peer bandwidth broker **(6)** is authenticated and the appropriate fields are modified and the RAA is sent to the next bandwidth broker in the chain back to the originating domain **(7)**. Internally to the domain, the bandwidth broker may modify traffic conditioners and PHB parameters in the ingress and egress border routers in the path of the flow (indicated by the green arrows in the figure). In addition, resource allocation internal to the domain may be initiated by the bandwidth broker. This would consist of modifying PHB parameters and traffic conditioners in internal routers.

- *Originating domain processing of the RAA*

  When the bandwidth broker of the originating domain receives the RAA **(7)** and authenticates it, the bandwidth broker completes any resource allocation actions within the domain, modifies PHB and traffic conditioner parameters at the egress router for the flow and forwards the RAA to the requesting end system **(8)**. This may include setting the marking functions for the flow in the access router serving the requesting end system (indicated by the green arrows in the figure).

  The end system receives the RAA and is able to send the flow. Note that there is nothing to prevent the end system from sending the flow earlier; however, the flow will not receive the requested service until the RAA is received and the DSCP of packets sent earlier than this will not be marked consistent with the service.

10.5.2.2 Case 2: Resource Request for Core Tunnel Services

In this section, the set-up of a pipe (core tunnel) between an origin domain and a destination domain is handled. Tunnel is a term used in this document for an inter-domain reservation where one or both ends of the reservation is not fully specified, and is not to be confused with IP tunnels or MPLS tunnels. It is a vehicle for aggregating reservations. A tunnel can extend from DS domain to DS domain.

This kind of request may originate in an end system that knows, for example, that it has a large number of requests for service of a certain kind to send to a destination domain and is prepared to aggregate the resource requests to intermediate domains. The request may also originate with a bandwidth broker, as a result of aggregation algorithms.

Core tunnels extend from the egress interface of the originating domain to the ingress interface of the destination domain. Note that tunnels as well as reservations are unidirectional. The setting up of a core tunnel involves the intermediate bandwidth brokers, but the use of it for aggregating individual flows does not.

There is a difference between the tunnels and the reservations. The tunnels have origin and destination pairs, while the reservations for several tunnels may be merged at the border router interfaces.

The establishment of a core tunnel is triggered in the origin bandwidth broker.

- *Behaviour in the originating domain.*

  The bandwidth broker in the origin domain creates an RAR which includes the IP prefix of the destination domain along with the normal information required in an RAR and an indication that a core tunnel is being requested. This RAR is sent to the bandwidth broker in the next domain in the path on the way to the destination domain.

- *Transit domain processing of the RAR*

  In all transit domains, except for the penultimate domain, the bandwidth brokers behave in exactly the same way as for an RAR with a fully specified destination address.

- *Penultimate domain processing of the RAR*

  In addition to all the checks outlined in the previous step, the bandwidth broker in the penultimate domain creates, on acceptance of the RAR, a *core tunnel voucher* which contains information about the reservation, ensuring that it fits within the SLS between the penultimate domain and the destination domain. This voucher is added to the RAR and sent to the destination domain. It is used later by the origin domain bandwidth broker to refer to the reservation.

  If the reservation is not accepted, the bandwidth broker returns an RAA.

- *Behaviour in the destination domain*

  When the bandwidth broker in the destination domain receives the RAR, it performs the following functions:

  a. Authentication that the request is indeed from a peer bandwidth broker.

  b. Checks that the RAR falls under the SLS with the sending domain connecting via the specified ingress router (interface).

  c. Checks that there are sufficient resources in the domain to support the RAR.

  d. Determination of whether the RAR can be accepted.

  If the outcomes of these decisions are positive, the destination domain bandwidth broker stores the voucher from the penultimate domain and stores also the identifier of the origin domain. It then returns an RAA to the penultimate domain.

  If the outcomes are negative, then it returns an RAA.

- *Transit domain processing of the RAA*

  In all transit domains (including the penultimate domain) the bandwidth broker authenticates the RAA from the sender and replaces the sender ID and authentication strings with its own ID and authentication string and then sends the RAA on to the following domain in the direction of the origin domain.

  At the same time, the bandwidth broker may make adjustments to traffic conditioning (shaping, policing, marking, metering) and PHB functions in its affected border routers and (possibly) in the internal routers of the domain.

- *Origin domain processing of the RAA*

On receiving the RAA for its request, the origin bandwidth broker authenticates the RAA and checks the information in it to see whether the request was accepted or not. If the RAR was accepted, the bandwidth broker stores the voucher created in the penultimate domain in the path. At this time, the bandwidth broker may also make adjustments to traffic conditioning and PHB functions in its border router, and it may at this time establish a TCP session with the bandwidth broker in the destination domain (if it has not already done so).

Other Tunnels

In addition to core tunnels, other configurations are possible, for example, where the source address is fully-specified (is an end system) but the destination address is not (*head tunnels*), or where the source address is not fully-specified but the destination address is (*tail tunnels*). Both of these cases can be handled with some minor modifications to this protocol (in the origin and destination domain BBs).

10.5.2.3 Case 3: Core tunnel handling of a request with fully-specified destination

In this case, the service request has a fully specified destination address, but a separate reservation in the core network(s) is not made. Instead this service request is aggregated into a *core tunnel* assumed in this case to be previously set up. Note that only the origin and destination bandwidth brokers and the end systems are involved in this communication.

- *Originating domain processing the RAR*

  The bandwidth broker in the origin domain receives an RAR from an end system in its control. According to its own algorithms, it chooses to aggregate this request with others in an existing core tunnel. The bandwidth broker checks the following:

  - Whether the requester is authorised for this service.
  - The route through the domain to the egress router. It was assumed that in setting up the core tunnel, the bandwidth broker would check to ensure that the resources to support it were available in the domain. However, that check could be delayed to, or repeated at this point.
  - Whether the flow fits in the core tunnel.
  - Whether the flow may be accepted for the specified service.

  If the outcomes of these decisions are positive, the bandwidth broker replaces the sender ID and authentication string in the RAR with its own ID and authentication string, and places the Core Tunnel Voucher TLV for the core tunnel into the message and sends the RAR directly to the bandwidth broker of the destination domain.

  If the outcomes are negative, then the bandwidth broker returns an RAA to the end system indicating failure along with a reason code, etc.

- *Destination Domain processing the RAR*

  When the destination bandwidth broker receives the RAR, it checks the following:

  a. Authenticate that the request is indeed from a peer bandwidth broker.
  b. Authenticate the Core Tunnel TLV
  c. Check that the requested resources fit in the core tunnel
  d. Determine the intra-domain route from the ingress router to the end system and decide whether the resources are available to support the flow.
  e. Determine whether the flow may be accepted (possibly according to the policies of the domain).

  In case these decisions have negative outcomes, an RAA is sent back.

In case all these decisions have positive outcomes, the bandwidth broker sends the RAR to the end system with appropriate changes. In this case, the destination end system makes the determination whether it can receive the flow. This is signalled with an RAA to the bandwidth broker of the destination domain. The RAA contains authentication of the end system, and parameters for the flow which the end system is willing to accept (which may be different from those received). In case the flow is rejected, the RAA contains a reason code and possibly hints about the set of service parameters that would be acceptable.

Upon receiving the RAA from the destination end system, the bandwidth broker authenticates the answer and forwards the RAA, with appropriate changes to the origin bandwidth broker. At the same time, the destination bandwidth broker may configure traffic conditioners at the ingress router and possibly at other routers along the intra-domain path to the destination Note: these are indicated by green arrows in the figure.

- *Origin processing of the RAA*

  When the bandwidth broker of the originating domain receives the RAA and authenticates it, the bandwidth broker completes any resource allocation actions within the domain, modifies PHB and traffic conditioner parameters at the egress router for the flow and forwards the RAA to the requesting end system. This may include setting the marking functions for the flow in the access router serving the requesting end system.

  The end system receives the RAA and is able to send the flow.

10.5.2.4 Releasing the reservation

Either of the endpoints of a reservation or the BBs in the endpoint domains may release the reservation. It is assumed that intermediate bandwidth brokers who are aware of a reservation (i.e. one representing a tunnel, not made within a tunnel) also know their peer bandwidth brokers both upstream and downstream with respect to the reservation. In the case that the reservation may have an exact end time, the reservation is removed automatically by all parties involved without the need for a takedown message to be sent.

10.5.3 RSVP

RSVP (the Resource ReServation Protocol) [RSVP] [RFC2205] is designed to provide end-to-end QoS signalling services for application data streams. Hosts use RSVP to request a specific QoS from the network for particular application flows. Routers use RSVP to deliver QoS requests to all routers along the data path. RSVP also can maintain and refresh states for a requested QoS application flow.

RSVP tries to fit well in the IntServ architecture with certain modularity and scalability. The design of the RSVP protocol distinguished itself in a number of fundamental ways, particularly, soft state management, two-pass signalling message exchanges, receiver-based resource reservation and separation of QoS signalling from routing.

The RSVP signalling model is based on a special handling of multicast. The sender of a multicast flow advertises the traffic characteristics periodically to the receivers via "Path" messages. On receipt of an advertisement, a receiver may generate a "Resv" message to reserve resources along the flow path from the sender. Receiver reservations may be heterogeneous. To accommodate the multipoint-to-multipoint multicast applications, RSVP was designed to support a vector of reservation attributes called the "style". A style describes whether all senders of a multicast group share a single reservation and which receiver is applied. The "Scope" object additionally provides the explicit list of senders.

Because the number of receivers in a multicast flow is likely to change, and the flow of delivery paths might change during the life of an application flow, RSVP takes a soft-state approach in its design, creating and removing the protocol states in routers and hosts incrementally over time. RSVP sends periodic refresh messages to maintain its state and to recover from occasional lost messages.

In the absence of refresh messages, the RSVP states automatically time out and are deleted.

The receiver in an application flow sets the desired QoS. To do this, the receiver issues an RSVP QoS request on behalf of the local application. The request propagates to all routers in the reverse direction of the data paths toward the sender. In this process, RSVP requests might be merged, resulting in a protocol that scales well when there are a large number of receivers.

Receiver-initiation is critical for RSVP to set-up multicast sessions with a large number of heterogeneous receivers. A receiver initiates a reservation request at a leaf of the multicast distribution tree, travelling towards the sender. Whenever a reservation is found to already exist in a node in the distribution tree, the new request will be merged with the existing reservation. This could result in fewer signalling operations for the RSVP nodes in the multicast tree close to the sender, but introduces a restriction to receiver-initiation.

RSVP messages follow normal IP routing. RSVP is designed to operate with current and future unicast and multicast routing protocols. The routing protocols are responsible for choosing the routes to use to forward packets, and RSVP consults local routing tables to obtain routes. RSVP is responsible only for reservation set-up along a data path.

RSVP carries the QoS data of the request through the network, visiting each node along the data path. To make a resource reservation at a node, the RSVP module communicates with two local decision modules, admission control and policy control. Admission control determines whether the node has sufficient available resources to supply the requested QoS. Policy control determines whether the user has administrative permission to make the reservation. If either check fails, the RSVP module returns an error notification to the application process that originated the request. If both checks succeed, the RSVP module sets parameters in a packet classifier and packet scheduler to obtain the desired QoS.

The definition of the required resources is not part of the RSVP standard, but commonly the IntServ specifications for Controlled Load and Guaranteed Services are used. RSVP allows for unicast and multicast reservations. Various filtering rules may be used to identify flows belonging to a reservation - commonly the 5-tuple is used.

RSVP scales in that it supports large multicast groups, at the cost of high complexity in dealing with multicast in its basic protocol. While the RSVP protocol is also able to make unicast reservations, it was designed specifically and optimally for multicast. This important RSVP design consideration leads to the fact that, even for unicast applications, a full-fledged set of features for supporting multicast is still needed

10.5.3.1 Extensions to RSVP

There have been various extensions to enhance the basic RSVP protocol: policy, cryptographic authentication, aggregation, tunnelling, refresh overhead reduction, diagnostics, RSVP-TE, DCLASS, null service, proxy, mobility schemes, etc. There has been a large amount of effort towards a global Internet QoS deployment based on RSVP since its development.

Only Standards Track RFCs and some Internet drafts listed below; informational and BCP RFCs (e.g., RFC2998) are not covered here.

[RFC2207] specifies an RSVP extension to use the IPSEC SPI (Security Parameter Index), in place of the UDP/TCP-like ports, so that data flows containing IPSEC protocols can be controlled at a granularity similar to that already specified for UDP and TCP. The IPv6 Flow Label can also be used as a key in the filters. Furthermore, reservations may be distinct or shared by several senders.

[RFC2996] introduces a DCLASS Object to carry Differentiated Services Code Points (DSCPs) in RSVP message objects.

[RFC2749] specifies the usage of COPS policy services in RSVP environments.

[RFC2380] presents the implementation requirements for running RSVP over ATM switched virtual circuits (SVCs).

[RFC2814] introduces an RSVP LAN_NHOP address object that keeps track of the next L3 hop as the PATH message traverses an L2 domain between two L3 entities (RSVP PHOP and NHOP nodes).

To provide sufficient information for debugging or resource management, RSVP diagnostic messages (DREQ and DREP) are defined in [RFC2745] to collect and report RSVP state information along the path from a receiver to a specific sender.

[RFC2746] describes an IP tunnelling enhancement mechanism that allows RSVP to make reservations across all IP-in-IP tunnels, basically, by way of recursively applying RSVP over the tunnel portion of the path.

To reduce the refresh volume and maintain reliability, [RFC2961] defines a Bundle message to reduce overall message handling load.

[RFC3175] allows to install one or more aggregated reservations in an aggregation region, thus the number of individual RSVP sessions can be reduced.

[RFC3209] specifies extensions to RSVP for establishing explicitly routed LSPs in MPLS networks using RSVP as a signalling protocol.

Section 5 of RFC3270 further specifies the extensions to RSVP to establish LSPs supporting DiffServ in MPLS networks, introducing a new DIFFSERV Object (applicable in the Path messages) and using pre-configured signalled "EXP<-->PHB mapping".

A detailed analysis of RSVP regarding multicast can be found in [Fu02].

The interactions of RSVP and Mobile IP have been well documented in [Thom01].

The security issues have been well analysed in [Tsch02].

RSVP needs to be developed more flexible and applicable for more generic signalling. RSVP proxies [BEGD02] extends RSVP by being able to originates or receive the RSVP message on behalf of the end node(s), so that applications may still benefit from reservations that are not truly end-to-end.

The Localised RSVP [MSK+02] draft presents the concept of local RSVP-based reservation that can be used to trigger reservation within an access network alone. In those cases, an end-host may request QoS from its own access network without the co-operation of a correspondent node outside the access network.

Below is a list of some other current Internet drafts related to RSVP.

- “An Evaluation on RSVP Transport Mechanism”, draft-pan-nsis-rsvp-transport-01.txt, Expires July 2003.
- “Towards RSVP Version 2”, draft-brunner-nsis-rsvp2-00.txt, Expires March 2003.
- “RSVP Path computation request and reply messages”, draft-vasseur-mpls-computation-rsvp-03, Expired Dec. 2002.
- “The Use of Bi-Directional RSVP in the Wireless Internet”, draft-shaheen-shahrier-nsis-brsvp-00.txt, Issued July 2002.
- “RSVP Security Properties”, draft-ietf-nsis-rsvp-sec- properties-00.txt, Expires April 2003.
- “Mobility Extensions to RSVP in an RSVP-Mobile IPv6 Framework”, draft-shen-nsis-rsvp-mobileipv6-00.txt, Expires January 2003.
- “Extended RSVP-TE for Point-to-Multipoint LSP Tunnels”, draft-yasukawa-mpls-rsvp-p2mp-00.txt, Expires June 2003.
- “RSVP-TE extensions for inter-domain LSPs”, draft-pelsser-rsvp-te-inter-domain-lsp-00.txt, Expires April, 2003.
- “Extended RSVP-TE for Multicast LSP Tunnels”, draft-yasukawa-mpls-rsvp-multicast-01.txt, Expires May 2003.

- "RSVP-TE Extension for Shared Mesh Protection", draft-liu-mpls-rsvp-shared-protection-00.txt, Expires April 2003.
- "Requirements for using RSVP-TE in GMPLS signalling", draft-matsuura-gmpls-rsvp-requirements-01.txt, Expired Dec. 2002.
- "Fast Reroute Extensions to RSVP-TE for LSP Tunnels", draft-ietf-mpls-rsvp-lsp-fastreroute-02.txt, Expires Aug. 2003.
- "A Proposal for RSVPv2", draft-westberg-proposal-for-rsvpv2-01.txt, Expires April 2003.
- "Using RSVPv1 as NTLP (NSIS Transport Layer Protocol): suggestions for modifications on RFC2205", draft-westberg-nsis-rsvp-as-ntlp-01.txt, Expires September 2003.

10.5.4 SIP

10.5.4.1 Overview

The Session Initiation Protocol (SIP) is a text based application-layer control (signalling) protocol for creating, modifying and terminating multimedia sessions [RFC3261]. It re-uses many of the existing Internet standards, such as DNS and e-mail style addressing of called/calling parties. SIP is a client/server protocol that is not tied to any particular lower-layer transport protocol. It is a layered protocol consisting of message encoding, transport, transaction and transaction user layers. In order to perform transactions it operates a challenge/response mechanism similar to HTTP.

The diagram in Figure 48 below shows the main components in a SIP connection. In the scenario shown, user to user signalling is routed via a SIP Proxy Server, which has similarities to Gatekeeper routed signalling in H.323, although a simpler model where SIP User Agents communicate directly is also possible. The SIP Proxy makes it easier to implement supplementary services such as redirection and follow-me.

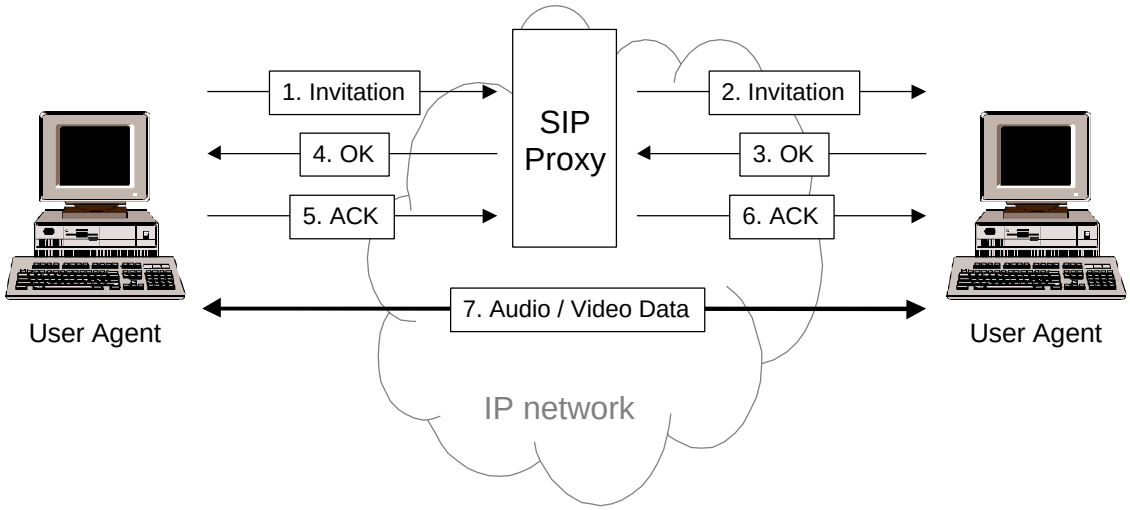


Figure 48 SIP Connection – the components

SIP operates by sending messages between the caller and the called, using e-mail style addressing (e.g. sip:bill@microsoft.com). Enough information is carried in these messages, which uses the Session Description Protocol (SDP), to allow full duplex multimedia communication after only one and a half round trips of signalling for a point-to-point session. SIP does not have the notion of dynamic logical channels as in H.323, and therefore commits from the outset to listening on a number of ports for various data streams which may or may not actually be used. Provision for conferencing is made by allowing either multicasting relations, a mesh of unicast relations, or a combination of both. As with H.323, this can be set up prior to communication taking place or on an ad hoc basis during a point to point call.

Audio/Video Data is carried in RTP media streams, which are routed directly between the User Agents.

10.5.4.2 Registrar

The Registrar is a server which accepts "Register" requests, allowing the network to keep track of the current location of users as well as their IP address (potentially dynamically allocated). The Registrar thus enables SIP to IP address mapping.

10.5.4.3 Proxies

A proxy server receives SIP requests and forwards them on behalf of the requester. In addition to DNS and Location service lookup, proxy servers may make routing decisions and enforce policy on where to send a request. A SIP request may traverse several proxies in order to reach its destination. Proxies may be also used either to transcode if the end points cannot agree on Codec negotiation. A proxy can be made to duplicate a request in order to try to contact several end points belonging to the same user. Other call agent functions can be implemented on the proxy such as call filtering and call redirection rules. Options are also available to ensure that all proxies traversed from the caller to the destination are also traversed on the way back. Domains may contain multiple proxies to service different user groups or provide different services.

10.5.4.4 SIP and Resource Reservation

SIP does not implicitly support end to end QoS. This is generally provided by use of a QoS reservation protocol such as RSVP in parallel with the SIP session negotiation [Camarillo02] [Sinnreich00][Johnston03]. The RTP stream is then carried over the RSVP session. Various methods of passing the QoS information necessary for reservation exist, from having RSVP capable UA to Server assisted QoS such as Q-SIP (see below). Aggregation of IntServ flows onto a DiffServ core network is performed as if SIP was not involved.

10.5.4.4.1 Q-SIP

Q-SIP [Veltri02] is an enhancement of the SIP protocol to carry end-to-end QoS related information. QoS extensions to the SIP message header are added by local servers, allowing use of existing SIP User Agents. Figure 49 shows an example architecture for Q-SIP. QoS aspects in the DiffServ core network are handled by the COPS protocol.

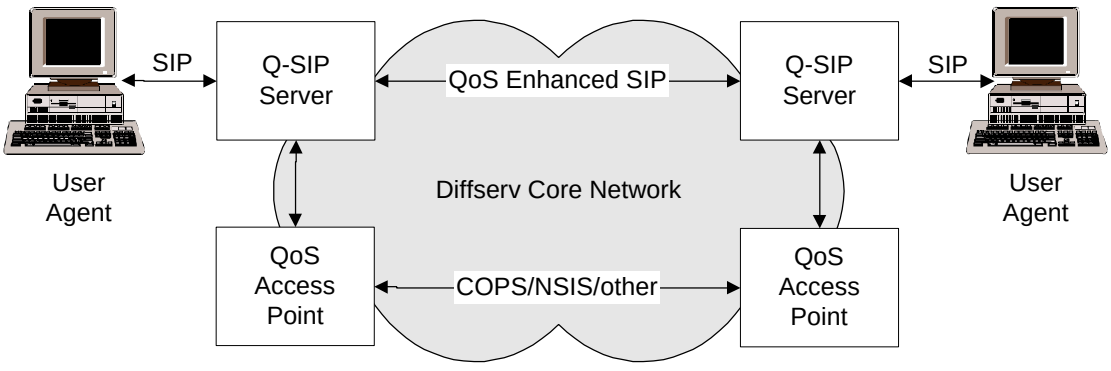


Figure 49 Q-SIP Network Diagram

A SIP session is established in the normal manner (Figure 49), the local Q-SIP server or proxy adding QoS information in a new SIP header, route information being added by transitory Q-SIP servers. The callee side Q-SIP server then has all of the information needed to request a specific QoS reservation across the core network to the caller Q-SIP server.

The QoS parameters such as the bandwidth and type of class are selected based on the type of media and codecs specified by the end UA and/or according to the user profile.

10.5.4.5 Suitability for MESCAL Signalling

SIP is a transaction-based protocol, similar to HTTP. It is extensible, allowing additional user defined messages. It provides no explicit support for aggregation, apart from what can be carried out by proxy servers based on message destination. It is not inconceivable that SIP could be used as a protocol for establishing aggregated media streams, though there is no real advantage in doing this and common usage tends towards COPS in the core network. SIP signalling takes place independently of the end-to-end RTP media stream, though it may use the same network links.

10.5.4.6 Related protocols

10.5.4.6.1 COPS-DRA

The Common Open Policy Service (COPS) [RFC2748] is a simple query and response protocol that allows policy decision servers (PDPs) to communicate policy decisions to enforcement points at network devices (PEP).

COPS-DRA (COPS Dynamic Resource Allocation) [Salsano01] is a new client type for the COPS protocol to support dynamic DiffServ resource allocation. The protocol carries the scope and amount of the reservation, the type of service required and flow identification information. COPS-DRA supports both outsourcing a provisioning views of resource allocation in both intra and inter-domain networks. In Figure 50 the ER acts as a PEP when communicating with the Bandwidth Broker. However, it assumes the PDP role when communicating with the QoS client. Thus, the same protocol is used at multiple levels of the hierarchy.

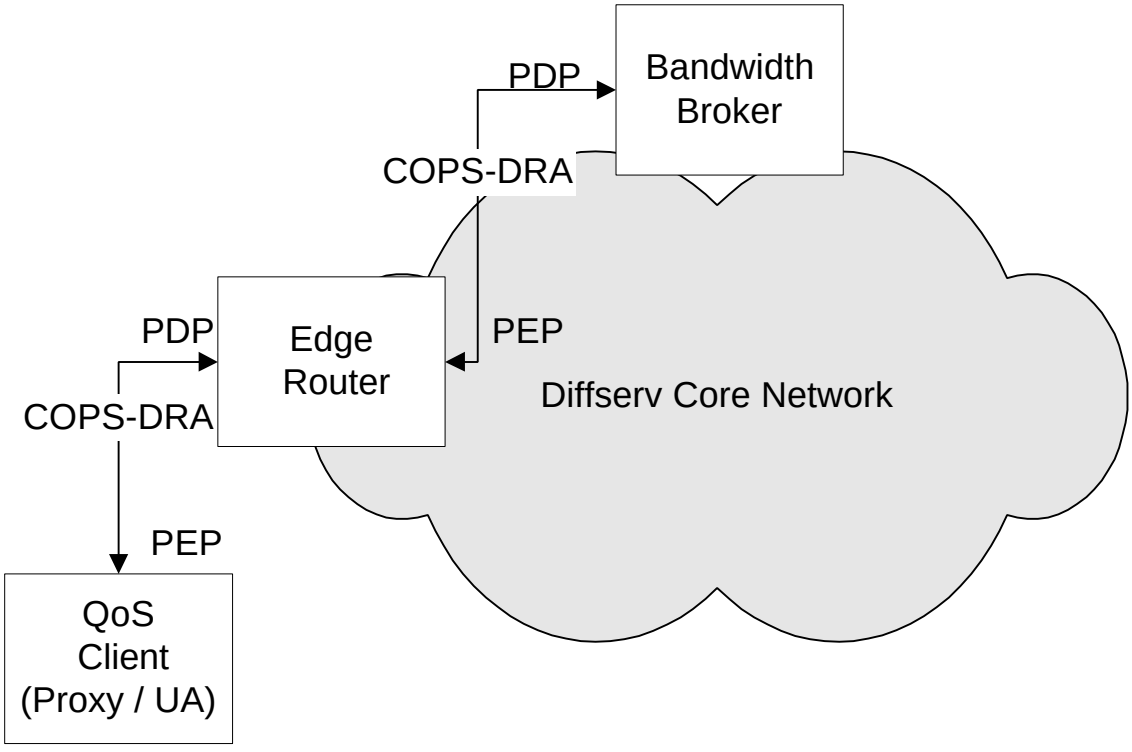


Figure 50 Use of COPS-DRA in a DiffServ Network

COPS-DRA can be used in conjunction with SIP / Q-SIP to provide QoS between SIP UA.

10.5.4.6.2 TRIP

Telephony Routing over IP (TRIP) [RFC3219] is a policy driven inter-administrative domain protocol for advertising the reachability of telephony destinations between location servers, and for advertising attributes of the routes to those destinations. TRIP is independent of signalling protocol and is modelled after BGP-4. However, it is enhanced with some link state features as in OSPF, and permits generic intra-domain LS topologies, which simplifies configuration compared to BGP. TRIP permits aggregation of routes as they are advertised through the network. A TRIP route is defined as the combination of a set of destination addresses, and an application protocol such as SIP.

10.6 Service Management

10.6.1 Overview

Broadly speaking, the term service management denotes the administration and management activities of a service provider related to services and customers.

Service management has been widely recognised as topic in its own right in the management of telecommunication networks. In the Telecommunications Management Network (TMN) model, proposed by ITU-T [TMN] as a means for organising the management of telecommunication networks, service management lies in the third layer of the TMN management hierarchy: below the business management layer, which is at the apex of the TMN model, and above the network management and network element management layers. Similar to TMN, other initiatives related to the management of telecommunication networks and services e.g. TINA [TINA], TMF [TMF] have also recognised service management as a distinct, but co-operating, layer in their architectures.

According to TMN, service management, as any other management layer, should encompass functions covering all aspects of management –the so-called FCAPS management functional areas: fault, configuration, accounting, performance and security management.

As the objective of service providers is to deliver value to their customers, the definition, design and automated execution of end-to-end process flows for providing services to customers are of paramount importance. By adopting a business process management view, TMF [TMF] approached this issue by defining and analysing three basic end-to-end processes common to any service oriented business: service fulfilment, service assurance and service billing. Figure 51 shows a broad breakdown of these end-to-end processes in terms of individual self-evident processes, which are arranged in a hierarchical TMN-like management structure; the network management layer corresponds to the lowest layer depicted in the figure, and the service management layer has been divided into the service development and operations and the customer care parts.

Alternatively to the TMN FCAPS management view stated above, Figure 51 shows the required functionality in the service management layer (cf. the upper two layers) from the perspectives of business processes.

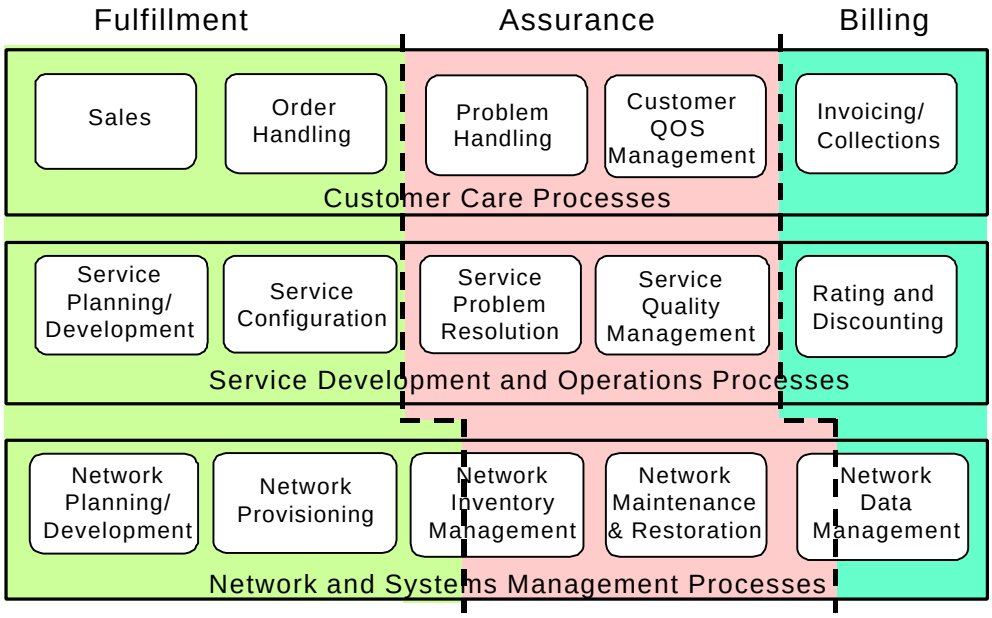


Figure 51: End-to-end process breakdown from service provider's business perspectives (source: TeleManagement Forum [TMF]).

10.6.2 Evolution of Work

Generally speaking, in the service management field, work in the literature, relevant standardisation bodies and international activities has been mainly devoted in the specification of appropriate architectures, informational and computational models and technologies for requesting and provisioning services. The definition of quality of service (QoS) and efficient means for provisioning QoS-based services has been the focal point of investigation.

In the early 90's, ITU-T [TMN] has set the basis for service management, laying down relevant principles and architectures. TMF [TMF] looked at service management issues from the perspectives of automating the flow of business processes from the standpoint of a service provider. TINA [TINA], adopting an 'all-software, out-of-the-box' approach, has proposed informational and computational architectures, spanning from user terminals to service and network providers, for initiating, controlling and managing service sessions for multimedia applications, including streaming. The IETF [26] has been mainly concerned with protocols for QoS service signalling, enhancements of network layer protocols for supporting QoS service requirements and models for providing unicast and multicast QoS-based services mainly from network perspectives -required support mechanisms at routers and protocol levels. The IST TEQUILA project [32] has specified a validated framework for provisioning QoS-based IP connectivity services, including models for describing relevant SLAs/SLSs and architectures, tools and algorithms enabling automated service subscription, invocation, configuration and graceful service delivery over DiffServ IP networks. Commercially available network management platforms, providing 'traditional' TMN-like FCAPS network management functions have all been enhanced to support QoS aspects at network, system and service levels.

Recently, with the prevalence of IP as the ubiquitous network layer and the advances made for supporting QoS-based services, efforts on service management aspects have become more intense. Given the multitude of services –for fixed and mobile users- the emphasis has now been shifted from computational architectures and protocols for requesting and provisioning services, to architectures and technologies enabling the fast introduction of services and the cost-effective management of their life-cycle; from service offering and configuration, to service operation, monitoring and billing. Service description, creation, execution, monitoring, accounting, billing, taking into account the diversity of the network and service environment and the different business relationships underlying service provisioning, are key aspects of current investigations. Service management technologies have

moved from complex platforms relying on multi-layer middleware, to tools and libraries relying on APIs (Advanced Programming Interfaces) based on widely deployed Internet-friendly programming technologies.

The above directions in service management are currently witnessed by the activities of a number of relevant industrial consortia and standardisation bodies. Typical examples of such works include: ETSI [ETSI] have specified OSA (Open Service Access), which defines an architecture that enables service application developers to make use of network functionality through open standardised interfaces; the OSA APIs. Aligned with OSA, the Parlay Group [Parlay] is an open multi-vendor consortium formed to develop open technology-independent APIs, essentially 'wrapping' services for fixed and mobile users, related protocols and management capabilities, to enable the development of applications and solutions operating across multiple networking platform environments. The Jain initiative [Jain], endorsed by a number of industry leaders, builds on Parlay APIs aiming at specifying a set of Java technology based APIs for enabling the rapid development of next generation telecom products and services on the Java platform. The World Wide Web Consortium (W3C) [W3C] has specified the so-called Web-Services, a suite of protocols and tools enabling the description, registration, discovery and access of services in the Web. Web-service architecture entails three standards: WSDL (Web Services Description Language), UDDI (Universal Description Discovery and Integration) and SOAP (Simple Object Access Protocol), and is based on Internet-friendly technologies like http/tcp and XML. The ipdr.org [IPDR] is an industrial consortium aiming at setting common specifications for IP-based services from accounting perspectives.

10.6.3 MESCAL Interest

MESCAL is concerned with the provisioning of QoS-based services in the Internet, across multiple provider domains. With reference to the process-map depicted in Figure 51, the areas of interest to MESCAL lie in the fulfilment (mainly) and assurance (to a lesser extent) aspects of the lower-two layers; the network management and service development and operations layers. Accounting and billing aspects fall outside the scope of the project.

QoS service provisioning is seen by MESCAL from network connectivity perspectives, rather than from the perspectives of higher-level, informational services. The work focuses on appropriate functions, mechanisms, tools and protocols at network and service layers required for ensuring the graceful delivery of the services across the Internet. As such, from service management perspectives MESCAL is primarily interested in the description and access of QoS-based Internet connectivity services and respective agreements -SLAs/SLSs, between customers and providers and between providers- and the configuration of the network infrastructure as appropriate as required for gracefully fulfilling the agreed services –interactions between service and network management layers and the network.

In the above set-up, of primary interest to MESCAL is the IETF work regarding service signalling and the TEQUILA QoS-based IP service framework, which are presented in more detail in the following sections. MESCAL is also interested, however at a lesser extent, in work -discussed in the previous section- related to fast service introduction, registration, discovery, access and API-based 'wrapping' of service capabilities, especially in Web-based technologies. The MESCAL interest in these works is mainly for better aligning its service-related work in the current trends of service management, thus increasing the applicability of the produced results. Last, it should be noted that the issue of QoS service provisioning, especially aspects related to service management, is at its infancy.

10.6.4 IETF QoS Service Models and Related Signalling Protocols

The enlargement of the Internet users community has generated the need for IP-based applications requiring guaranteed Quality of Service (QoS) characteristics. To this end, the IETF has proposed the Integrated Services (IntServ) [RFC1633] and Differentiated Services (DiffServ) frameworks.

The Integrated Services approach (IntServ) relies on explicit request and reservation of resources for individual flows, with the Resource ReSerVation Protocol (RSVP) [RFC2205] used to signal the required QoS characteristics. While IntServ operates on a per-flow basis and hence provides a strong

service model that enables strong per-flow QoS guarantees, it suffers from scalability problems due to the large amount of flow state information that needs to be maintained in core network routers. On the other hand, DiffServ was conceived to provide QoS in a scalable fashion. Per-flow information is kept only at the edge of a domain and flows are aggregated into a limited set of traffic classes within the network, resolving the scalability problem at the expense of looser QoS guarantees per flow.

In addition to best-effort service, two QoS service models are prescribed by the IETF work: the Expedited Forwarding (EF) model for providing *quantitative* guarantees up to certain bandwidth limits, for example for Virtual Wire and other guaranteed QoS services; and, the Assured Forwarding (AF) model for providing coarser-grained quantitative QoS guarantees up to certain bandwidth levels, beyond which *qualitative* QoS guarantees can only be given. For multicast services, the IETF prescribes for a destination-driven, source-specific model [SSM], whereby the destinations initiate/terminate their request to join a known multicast group from a given source; destinations may have different QoS requirements.

Several QoS signalling mechanisms have been defined in the above QoS Internet frameworks.

To overcome the RSVP scalability problems, the use of a single RSVP reservation to aggregate other RSVP reservations across a transit routing region has been proposed [RFC3175]. It proposes a way to dynamically create the aggregate reservation, classify the traffic for which the aggregate reservation applies, determine how much bandwidth is needed to achieve the requirement, and recover the bandwidth when the sub-reservations are no longer required. Moreover, a number of mechanisms have been suggested that can be used to reduce processing overhead requirements of refresh messages, eliminate the state synchronisation latency incurred when an RSVP message is lost and, when desired, refresh state without the transmission of a whole refresh messages [RFC2961].

BGRP (Border Gateway Reservation Protocol) is a candidate protocol for inter-domain aggregate resource reservation [Pan00], described in Section 10.5.1. It operates across ASs, leaving each stub or transit domain free to use its own intra-domain reservation protocol. BGRP builds a sink tree for each of the stub domains that aggregates bandwidth reservations from all data sources in the network and it bundles all the reservation messages into a single periodic refresh.

In addition to network-layer protocols for (connectivity) QoS-services, IETF initiatives have specified protocols for QoS-service support at transport and application layers. The RTP/RTCP (Real-Time Control Protocol) protocol [RFC1889] provides for end-to-end network transport and related control functions suitable for applications transmitting real-time data, such as audio, video or simulation data, over multicast or unicast network services. The SIP protocol (Session Initiation Protocol) [RFC2543] is a text-based protocol, similar to HTTP and SMTP, for initiating interactive communication sessions between users; such sessions may include voice, video, chat interactive games etc. SIP prompts for a clear-cut between the interactions required between IP connectivity flow and applications multimedia sessions. However, its applicability has serious constraints due to scalability and processing overhead. Similar to SIP, the H.323 protocol has been proposed by ITU-T for initiating multimedia teleconferencing sessions over the Internet. It utilizes the RTP/RTCP from the IETF, along with internationally standardized codecs and can apply to multipoint and point-to-point sessions.

While the signalling approaches described above have assumed a layer 3 (and above)-based approach, significant work has been done in the area of signalling between management entities called Bandwidth Brokers (BB), responsible to manage resources within a QoS domain [RFC2638]. The Internet2 Qbone signalling design team has developed the requirements for an inter-domain bandwidth broker protocol and published the initial draft of the protocol called SIBBS (Simple Inter-domain Bandwidth Broker Signalling) [Qbone], described in Section 10.5.2. SIBBS is a request-response protocol between the BB peers that carries the essential information for requesting a service and answering with admission control decisions for aggregates and exchange traffic.

A new working group has been formed in IETF to develop the requirements, architecture and protocols for the next generation of signalling, called NSIS (Next Steps In Signalling) [NSIS]. The first requirements draft is currently being discussed for the NSIS signalling protocol, considering general cases of QoS signalling based on different scenarios, both wired and wireless (mobile IP)

[Partain02]. This work is of particular interest to MESCAL, which intends to contribute to the NSIS working group by specifying SLS semantics and parameters together with signalling protocols for SLS negotiation and invocation.

Another initiative worth mentioning is the ETSI-TIPHON, which addresses the challenges of the interoperability of IP telephony systems with switched circuit networks. The main contribution of this initiative is its valuable generic architecture describing the general interfaces and interactions between transport and application plane. However, it is restricted to voice only, and –within the architecture- the IP transport network is handled as a black box, omitting all interactions between service and resource management. For example, this architecture has major consequences for the BGP-routing, which remain unclear. The architecture remains moreover at the pure theoretical level without any practical verification.

10.6.5 The TEQUILA QoS Service Management Framework

10.6.5.1 A Hierarchical QoS Service Model

A hierarchical service model is adopted (Figure 52), which spans from Service Level Agreements (SLAs) to Per Hop Behaviours (PHBs), the basic QoS building block in IP DiffServ networks. SLAs describe all aspects of a service contract. The technical aspects of a service contract are described by the so-called Service Level Specifications (SLSs). For QoS-based IP connectivity services SLSs are modelled on the basis of standard templates proposed by TEQUILA (section 10.6.5.2).

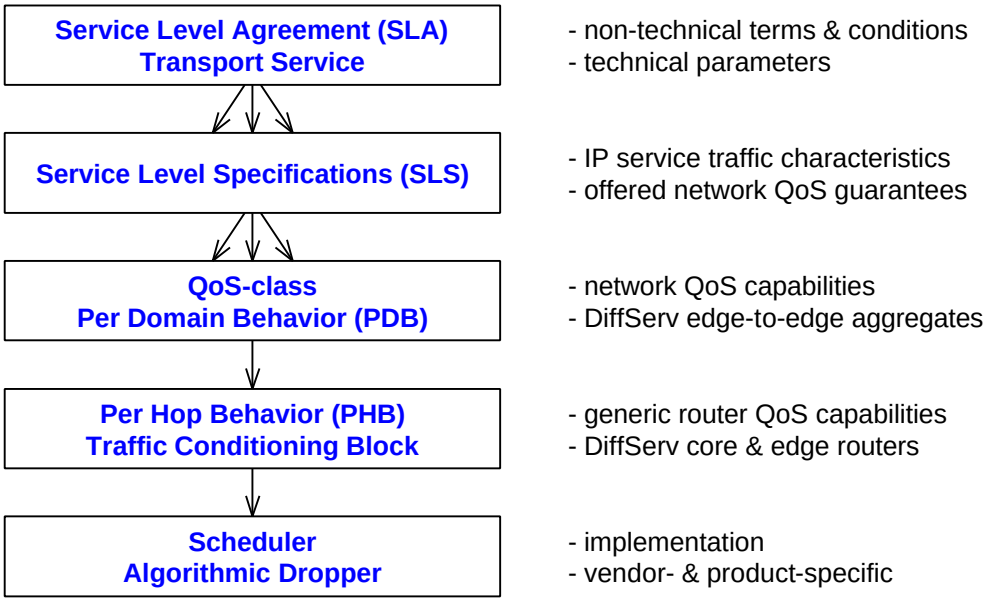


Figure 52: Hierarchical service model

The service hierarchy introduces the notion of *QoS classes* to link SLSs with PHBs. The notion of the QoS class is introduced to substantiate this mediation. QoS classes expose the elementary network-wide QoS transfer capabilities and they are bound to the specific technology employed and capabilities provided by the network. In IP DiffServ networks, QoS classes consist of an Ordered Aggregate (OA) and associated QoS parameters such as one-way delay and packet loss. Each service corresponds to a number of SLSs, and each SLS corresponds to a number of QoS classes. Therefore, given a service, its QoS is completely defined through the QoS classes of its constituent SLSs. For example, a Virtual Wire (VW) QoS class could be defined to denote an edge-to-edge transport capability with a guaranteed maximum packet delay and a guaranteed throughput for an aggregate IP packet stream marked as Expedited Forwarding (EF). QoS classes should be seen as specifications of Per Domain Behaviours. We have adopted the following specification of a QoS class.

Parameter	*Description*

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

Table 7: Specification of a DiffServ QoS class

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

Networks can support certain QoS classes through deploying dedicated Traffic Conditioning Block (TCB) at the edge routers, PHBs throughout the network interfaces, and an overall resource management system. Supporting customer specific SLSs boils down to a "service mapping" of the SLS to the corresponding QoS classes and SLS admission control blocks, while the network should be suitably engineered to gracefully sustain the traffic of the admitted SLSs.

10.6.5.2 SLS Template

According to the IETF DiffServ working group, a Service Level Agreement (SLA) is "*the documented result of a negotiation between a customer and a provider of an IP service that specifies the levels of* availability*, serviceability, performance, operation or other attributes of the transport service*" [14]. The SLA contains technical and non-technical terms and conditions. The technical specification of the IP connectivity service is given in Service Level Specifications (SLSs). A SLS "*is a set of technical parameters and their values, which together define the IP service, offered to a traffic stream by a DiffServ domain*". SLSs describe the traffic characteristics of IP flows and the QoS guarantees offered by the network to these flows.

The DiffServ working group does not intend to specify further the content of a SLS beyond the loose definitions given above. Nevertheless, the definition of a SLS is a key-step towards the provisioning of value-added IP services because it specifies the semantics of the interface between the provider and the customer and between providers, i.e. *the technical terms and conditions*. Standardisation of SLSs is also necessary to allow for highly developed levels of dynamic negotiation of service contracts and service provisioning. Moreover, the design and the deployment of Bandwidth Broker capabilities require standardised SLS semantics.

To the above end, TEQUILA has proposed a standard template for describing the parameters and semantics of SLSs for QoS-based IP services. The basic parameter groups of the SLS template with a brief description are presented in Table 8. More details can be found in [Goder02], [TEQ-SLS].

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

Table 8: The TEQUILA SLS Parameters

As it can be seen from the above table, the contents of SLSs include the essential QoS-related parameters: topological scope and flow identification, traffic conformance parameters and service guarantees. Note that although a number of performance and reliability parameters may be specified, in practice a provider would only offer a finite number of services, even for those with quantitative QoS guarantees. This approach simplifies the TE problem from the providers' perspective.

Note that a SLA may contain a set of SLSs. The above definition of a SLS is uni-directional, thus requiring two symmetric SLSs to describe services such as a bi-directional Virtual Leased Line (VLL) or a telephone call.

The following discuss further the identified SLS parameters.

The *Scope* of an SLS associated to a given service offering uniquely identifies the geographical and 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 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 Description* (*FlowDes*) of an SLS associated to a given service offering indicates for which IP packets the QoS policy for that specific service offering is to be enforced. A SLS has only one FlowDes, which can be formally specified by providing one or more of the following attributes: DiffServ information, source information, destination information, application information. The FlowDes provides the necessary information for classifying the packets at a DS boundary node. The packet classification can either be Behaviour Aggregate (BA) or Multi-Field (MF) based.

Traffic Envelope describes the traffic characteristics of the IP packet stream identified by FlowDes in order to receive the treatment indicated by the *Performance Parameters* (see below). These parameters are fed to the traffic conformance blocks at the edge of the network to uniquely identify the "in-profile" and "out-of profile"[2] (or excess) packets of an IP stream entitled to receive the specific QoS. The following is a non-exhaustive list of potential conformance parameters: *peak rate p* in bits per sec

[2] Note that the conformance result might not necessarily be of a binary mode (in/out) but it could also be multi-level (e.g. using a Two-rate Three-colour Marker algorithm).

(bps), *token bucket rate r* (bps), *bucket depth b* (bytes), *minimum MTU* - Maximum Transfer Unit - *m* (bytes) and *maximum MTU M* (bytes).

Excess Treatment describes how the provider should process the excess traffic, i.e. the out-of-profile traffic. 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 shaper's buffer size for shaping.

The *Performance Parameters* describe the packet transfer guarantees the network should offer to the customer for the packet stream described by the FlowDes, over the geographical/topological extent given by the *Scope*. There are four performance parameters: *delay, jitter, packet loss, and throughput.*[3] These parameters are specified as worst-case (deterministic) bounds or as quantiles. Delay, jitter and packet loss apply only to in-profile traffic. Throughput is the rate measured at the egress. Performance parameters might be either 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 for delay and packet loss may be "qualified". Possible qualitative values for delay and/or loss could be: *high, medium, low.* The actual "quantification" of the relative difference between high, medium and low is a policy-based decision (e.g. high = 2 x medium; medium = 3 x low). If the performance parameters are not quantified nor qualified the service will be best effort.

Service Schedule indicates the period of time the service can be 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.

Other parameters could also be specified, such as: *Reliability* indicating the maximum allowed mean downtime per year (MDT) and the maximum allowed time to repair (TTR) in case of service breakdown; *Assurance Level*, indicating the percentage of the time by which the provider must be able to conform to the specified SLS parameters.

[3] For each of these parameters we must specify a *time interval* and in some cases (e.g. delay) a quantile.

	Virtual Leased Line Service	Bandwidth Pipe for Data Services	Minimum Rate Guaranteed Service	Qualitative Olympic Services		The Funnel Service
Comments	The following is an example of a uni-directional VLL, with quantitative guarantees	Service with only strict throughput guarantee. TC and ET are not defined but the operator might define one to use, for protection.	It could be used for a bulk of ftp traffic, or adaptive video with min throughput requirements	They are meant to qualitatively differentiate between applications such as:		It is primary a protection service, restricts the amount of traffic entering a customer's network
				on-line web-browsing	e-mail traffic	
Scope	(1|1)	(1|1)	(1|1)	(1|1) or (1|N)		(N|1) or (N |all)
Flow Description	EF, S-D IP-A	S-D IP-A	AF1x	MBI		AF1x
Traffic Conformance	(b, r) e.g. r=1	NA	(b, r)	(b, r), r indicates a minimum committed Olympic rate		(b, r)
Excess Treatment	Dropping	NA	Remarking	Remarking		Dropping
Performance Parameters	D =20 (t=5, q=10e-3), L=0 (i.e. R = r)	R = 1	R = r	D=low L=low (gold/green)	D=med L=low (silver/green)	NA
Service Schedule	MBI, e.g. daily 9:00-17:00	MBI	MBI	MBI	MBI	MBI
Reliability	MBI, e.g. MDT = 2 days	MBI	MBI	MBI	MBI	MBI

(b, r): token bucket depth and rate (Mbps), p: peak rate, D: delay (ms), L: loss probability, R: throughput (Mbps), t: time interval (min), q: quantile, S-D: source & destination, IP-A: IP address, PN: port number, MBI: may be indicated, NA: not applicable

Table 9 Example SLS parameter settings for various services.

10.6.5.3 Service Negotiation Protocol (SrNP)

Compared to manual service negotiation methods, through fax or email for instance, automated service negotiation offers a high degree of flexibility to the customer and provider by reducing the time to request and gain access to services. To this end, TEQUILA specified a protocol for SLS negotiation, the *Service Negotiation Protocol (SrNP)*.

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

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

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

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

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

Figure 53 depicts alternative protocol stacks for realising SrNP. SrNP messages could be encoded in ASCII, BER/TLVs or XML as convenient for the stack used. Note also that it could be possible to encapsulate SrNP messages in widely deployed protocols such as RSVP (by defining new TLVs) and COPS (by specifying a new client-type). The latter is required when SrNP is to be used at invocation times. Currently there are two implementations of SrNP; one based directly on TCP/IP and the other on HTTP. In both implementations, the SrNP messages as well as the SLA and the revised alternatives were encoded in XML.

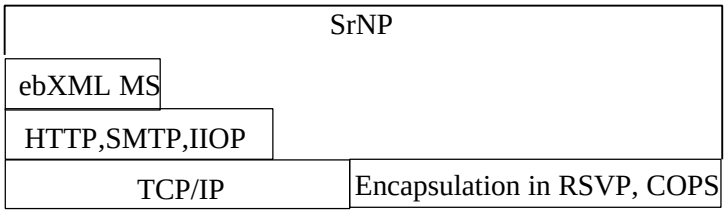


Figure 53: SrNP Protocol Stacks

10.7 Service Admission Control

10.7.1 Overview

The enlargement of the Internet user community has generated the need for IP-based applications requiring guaranteed Quality of Service (QoS) characteristics. The Integrated Services (IntServ) and Differentiated Services (DiffServ) frameworks have been proposed to address QoS. While IntServ operates on a per-flow basis and hence provides a strong service model that enables strong per-flow QoS guarantees, it suffers from scalability problems. On the other hand, DiffServ keeps per-flow information only at the edge of a domain and aggregates flows into a limited set of traffic classes within the network, resolving the scalability problem at the expense of looser QoS guarantees.

Beyond the standardised functionality at the IP layer, a large body of work has been devoted to architectures and functions necessary to deliver end-to-end QoS. These functions can be categorized into Traffic Engineering (TE) functions and Service Management (SrvMgt) functions [Trimin01], [Tequila]. TE functions are mainly concerned with the management of network resources with the purpose to accommodate offered traffic in an optimal fashion. SrvMgt functions deal with the handling of customer service requests, trying to maximize incoming traffic, in terms of number of contracts and throughput, while respecting the provider's commitments on the agreed QoS guarantees. SrvMgt mechanisms for service offering, agreement and activation need to be in place. In addition, in order to guarantee the agreed QoS requirements, SrvMgt needs to avoid overloading the network,

beyond loads that it can gracefully sustain. SrvMgt functions that deal with the latter task are referred to as service admission control.

Service Admission control, as placed above, is a key component for QoS delivery in IP networks as it determines the extent to which network resources are utilized, while ensuring that the contracted QoS characteristics are actually delivered.

10.7.2 Admission Control Schemes

The issue of service admission control in QoS-based multi-service networks, being an important aspect for QoS delivery, has received a lot of attention in the literature.

Admission control schemes usually operate at service invocation request epochs (call control).

Inherent to admission control is the well-know trade-off between network utilisation and QoS deterioration; the more traffic is injected in the network, the higher the utilisation of the network resources, but the more the likelihood of deteriorating the QoS of the traffic delivered by the network.

It should be noted that most of the noted admission control schemes aim at ensuring statistical (not hard) QoS guarantees. The provisioning of hard QoS guarantees implies the adoption of worst-case models for characterising the behaviour of traffic sources and the aggregation of traffic streams, and peak allocation-based reservation schemes for the allocation of network resources. However, for admission control, even peak allocation schemes cannot provide hard guarantees, unless complete partitioning of network resources in the core and hard edge-to-edge reservations are applied, which are regarded to be prohibitively costly, especially for bursty traffic. Moreover, statistical/stochastic models cannot provide hard QoS guarantees, because of the assumptions and approximations pertinent in modelling and aggregating the largely unpredictable user traffic.

Different admission control schemes differ in the way they tackle the following issues pertinent to the design of any admission control scheme:

- Place where admission decisions are taken e.g. centralised or distributed at ingress and/or egress points and/or end-systems.
- Information they utilise e.g. passive, active measurements, network state indications, pre-allocated network capacity with respect to certain QoS traffic.
- Logic/model they employ/rely for asserting network availability to sustain QoS, based on available information.
- Fairness criteria they target at e.g. at different QoS traffic and/or ingress/egress levels.

From the reported work in the literature and the approaches emerging from the IETF, the main schemes for service admission control are [Sargen]:

- RSVP signalling,
- bandwidth broker-based,
- dynamic packet state,
- aggregation in IntServ,
- measurement-based,
- end-point control through probing, and
- model-based.

The above approaches for admission control are described in the following.

RSVP signalling

The RSVP (resource ReSerVation Protocol) is a protocol for establishing and maintaining resource reservations in IntServ networks. It aims to communicate the resource demands and reservations to

each router along the flow's path. Although this signalling protocol is very strong in providing QoS support, it is not scalable, since it is necessary to maintain a flow state in each router along the flow's path, and all routers participate in the signalling protocol. The number of RSVP messages processed is proportional to the number of flows in the network and bandwidth must be reserved in each router on a per-flow basis. Both these disadvantages can lead to poor router performance.

Bandwidth-broker-based admission control

Bandwidth Brokers (BB) remove the need for QoS reservation states in the core routers, by centrally storing and managing this information. The main modules of the BB are the call admission control and routing ones. The former maintains the QoS state of the network domain and is responsible for admission control and resource reservations. The latter decides the path that the admitted flow will traverse towards the receiver. The general description of the call admission control module is as follows. When a new flow with specific traffic parameters, delay and loss requirements requests admission, it sends a QoS request message to the BB. The BB recalculates the available bandwidth in each link, and verifies if there is a path where the new flow can be admitted or not. If the flow is admitted, the BB sends a message to the sender with a positive answer to the flow's request, and updates its database. The available bandwidth in each connection is calculated through information stored in the BB about active flows, their traffic characteristics and their paths. Flows with the same characteristics may be grouped in service classes, such that the BB operations become faster and the number of requested flows that a BB can support increases. Although in this architecture the core routers will be freed from performing admission control decisions, the BB needs to manage the overall network and to store information about all elements, flows and paths in the network. This is very hard even for only one element. Therefore, for large networks, a distributed mechanism could be better employed.

Dynamic packet state (DPS)

In the DPS [Stoica99] technique, the flow state information (like reserved rate, variables used in the scheduling process) is inserted into packet headers, which overcomes the need for per-flow signalling and state management. The ingress router initialises the state information. Core routers process each incoming packet based on the state carried on it and eventually update its internal state and the state in the packet's header before forwarding it to the next hop. This mechanism uses core stateless scheduling disciplines [4], which calculate the packet's deadline, based only on the state variables of the flow it belongs to. At core nodes packet classification is no longer needed and packet scheduling is based only on the state carried in packet headers. Thus, per flow state can be stored only in the ingress node and the core nodes retrieve it in each core node. In terms of admission control, RSVP signalling is used to communicate between the sender and receiver, but RSVP messages are only processed by edge nodes. Upon receiving this signalling message, each node along the path performs a local admission control test based on the aggregate reservation rate in that node. With this technique, the core routers are freed from maintaining per flow state, but a deterministic service is provided since the admission control is based only on the flow's rate inserted in the packet header. This reduces the utilisation. Moreover, it is required that all routers in the flow's path implement the same scheduling discipline.

Aggregation in IntServ

Aggregation [Baker00] is a mechanism used to reduce the number of signalling messages in the IntServ QoS architecture. In this technique the admission control is only performed on an aggregated set of flows and therefore core routers need only to maintain the reservation state of each aggregate. The RSVP protocol is used, but only for aggregate flows. Thus, the signalling and the amount of stored state information in the core routers can be highly reduced. The aggregation implies the following trade-off: with more aggregation, more flows are not admitted and the utilisation decreases; with small aggregation the decrease in utilisation is neglected but the number of signalling messages remains high.

Measurement-based admission control

In this scheme [Centink], admission decisions are taken by ingress/egress routers and are based only on collected aggregate measurements reflecting the statistics of existing flows. The key technique is to passively measure on-line the available service in the end-to-end path, that is, to estimate the *available bandwidth* at all network bottlenecks along the connection's end-to-end route. Using a "black box" system model, the measurements can incorporate the cross traffic effects without explicitly measuring it or controlling it. Cross traffic is the traffic that is merged in some links with the traffic that is being measured. Since the peak rate of connections is usually known, a simple admission control criterion is for the new connection's peak rate to be less than this available bandwidth end-to-end. Alternative criteria could also be used e.g. statistical estimations of requested traffic or the user traffic descriptors as specified in the SLSs. Although such schemes do not require maintenance of per-flow state information neither in core nor in edge routers, they are hindered by estimation errors, coarse prediction of the dynamics of the system and memory related issues. Network load conditions may change and the QoS requirements may be degraded.

End-point admission control through probing

In this scheme [Elek00] the admission of a new flow is performed by the end-hosts or egress/ingress routers through the inference of the network congestion state in the flow's path. Before a new flow is established, the sender sends a packet stream to the flow's path with the same traffic characteristics of the flow that is requesting admission. The packet loss ratio, the delay or delay variation, are measured at the receiver, which verifies the network congestion level. This is called probing. If the measured performance is acceptable (according to the required service QoS), the flow is admitted; otherwise it is rejected. The QoS functionalities in this mechanism are pushed to the end-points, precluding the need of a signalling protocol or special functions in the core or edge routers. The overhead introduced with active probing and the set-up time required to initiate a call are the main disadvantages of this technique.

Model-based admission control

Model-based approaches maintain state information for active services and employ mathematical models for injected traffic. Along these lines, several algorithms have been proposed, utilising different mathematical models and service state information so as to determine whether or not new traffic flows can be admitted to the network without affecting the QoS characteristics of the existing flows.

10.7.3 Work Survey

The following present a non-exhaustive list of admission control solutions noted in the literature based on the above schemes:

- A novel Bandwidth Broker architecture for scalable support of guaranteed services [Zhang]

The BB architecture and the DPS techniques are utilised. A policy control module is introduced to configure the admission control and routing modules of BB. A path-oriented approach is adopted to improve performance against traditional hop-by-hop schemes. Flows are aggregated to macroflows but the dynamic aggregation (microflow joining and leaving macroflows at any time) is recognized as an unsolved problem.

- Endpoint admission control: architectural issues and performance [Bresl00]

The end-point probing mechanism is utilised. Fair queuing scheduling is excluded; instead, strict priority is used to avoid false probing results in cases of borrowed bandwidth from other classes. The trashing problem is addressed. This is the case when flows arriving rate is high enough to result to probing overlaps.

- Modelling the performance of distributed admission control for adaptive applications [Bain]

The end-point probing mechanism is utilised. Each flow is restricted to transmit using a specific rate from a defined set of rates, and can switch between the rates of that set. The decision is based on probing during invocation and during transmission times.

- Distributed admission control [Kelly]

The end-point probing mechanism is utilised. A trade-off between connection blocking and packet loss is recognised. Congestion detection is preformed by every hope down the path, that the probing packets traverse, and these packets are marked accordingly. The high importance of the timescale of measurements and the low importance of history is argued.

- Admission control based on end-to-end measurements [Elek00]

The end-point probing mechanism is utilised. The controlled-load service (CLS), that provides network state with bounded and well-known worst-case behaviour (primary developed for real time applications) is examined. The thresholds of admission control are specified in the CLS contract. This service requires its own capacity partition on the network links. Probing traffic is transmitted with lower priority than service traffic. Small buffers are used to limit delay and only loss is considered. In the case of invocation blocking source backs off random time before retransmitting.

- PBAC: Probe-Based Admission Control [Ivars]

The end-point probing mechanism is utilised. In addition to the above mentioned CLS the guaranteed service is examined (no Loss). The acceptance threshold is fixed for the service class and is the same for all sessions. The class definition should also state the maximum data rate allowed per session, which should be kept low, to ensure that statistical multiplexing works. It is stated that there is a proven linear relationship between probe and session loss, not dependent on the type of sources. A problem is recognised when sessions exist that transmit nothing for a period longer than the probe length. A proposed solution is to keep all sessions bandwidth smaller than 1% of the link capacity in order for multiplexing to produce reliable results.

- The DIY Approach to QoS [Karlsson]

The end-point probing mechanism is utilised. Three classes of service are assumed, guaranteed service, controlled-load service and best effort. This number of classes is believed to be necessary and sufficient. Two different priorities must exist for the each class (except BE), the lowest one is used by probing. Each router is responsible for holding only one state variable per output link, the amount of reserved capacity per link.

- Egress admission control [Cetink]

The measurement-based admission control mechanism is utilised. Coarse control is performed at edges using limited information for incoming traffic and network status. The network itself and the sharing of resource between classes is treated as a black box. The admission decision is based on service measurements from all edges, using differently congested paths and on accumulated input/arrival measurements.

- Implicit admission control [Mortier00]

The measurement-based admission control mechanism is utilised. It is argued that the network should perform admission control, since sources in competition for a resource cannot cooperate so it is up to the provider to set the admission thresholds. Measurements for making estimates that influence the admission decision are undertaken on aggregates and periodically, not on connection arrival times. TCP retransmissions are eliminated, therefore flows conclude successfully faster, hence more flows are admitted. Utilisation is kept high due to TCP greedy nature.

- Utilisation-based admission control for real-time applications [Xuan]

The measurement-based admission control mechanism is utilized. The worst-case achievable utilisation is defined (and used as reference) as the utilisation level below which all the workload using the resource is guaranteed to meet its performance targets. Admission control is made scalable by using a configuration-time test to determine a safe utilisation level (a-priori). Admission control at run time then is reduced to simple utilization tests on the servers along the path of the flow. A number of resource management components are required for materialising the above concepts; the Configuration, Run-Time Admission Control and Packet Forwarding components.

- Distributed connection acceptance control for a connectionless network [Gibbens99]

The measurement-based admission control mechanism is utilised. Intelligence is delegated to the end-systems (named as gateways). The gateways determine the network's congestion state based on simple marking and pricing schemes provided by the network. The call blocking probability is estimated by each gateway based on the time the specific gateway is out of the admissible area. Thresholds are differentiated per gateway.

- Admission control for statistical QoS: theory and practice [Knight99]

The model-based admission control is utilised. Several algorithms for admission control are presented: Average and peak rate combinations, additive effective bandwidth, engineering the 'loss curve', maximum variance approaches, refinements of effective bandwidth using deviations theory. The ultimate goal of the admission control algorithm test is, identified to be, the correct determination of the admissible region. It is argued that measurement-based approaches do not provide statistical QoS guarantees.

- End-point admission control: network-based approach [Choi01]

The model-based admission control is utilised. Signalling-free admission control schemes are defined as these schemes that are limited to the edge of the domain. These signalling-free schemes are divided into two categories, the host based and the network based, depending on the point where admission control is made. An off-line pre-allocation mechanism is adopted, for resources to minimise overload and latency at invocation times. Lightweight mechanisms to adjust initial pre-allocation are considered.

- Adaptive connection admission control for mission-critical real-time communication networks [Devalla]

The model-based admission control is utilised. QoS ranges are used, instead of specific values. A QoS adaptation mechanism is incorporated to offer the best possible QoS to connections contingent to the available resources. Connections are classified based on their criticality level and treated during admission control according to that level.

- A feedback-based, two-level, admission control for providing QoS in DiffServ IP networks: The TEQUILA approach [Mykon03]

Departing from the view that the ability of the network to sustain QoS cannot be safely guaranteed a priori (cf. discussion in section 7.4), either using a model-based or a measurement-based approach, the scheme relies on a feedback-based model for asserting the risk of QoS deterioration. This feedback is at two levels of abstraction: on the ability of the network to deliver QoS, determined through the off-line traffic engineering functions that dimension the network on the basis of anticipated demand and on the actual status of the network to deliver QoS, provided through measurements. Based on this feedback, the proposed scheme assesses the risk of QoS deterioration and accordingly adjusts the parameters for admitting service requests.

Admission control logic may be applied at both service subscription and service invocation epochs. At service subscription epochs, the admission control determines whether to accept a requested subscription or to initiate negotiations, based on the long-term ability of the engineered network (cf. the availability matrix produced by TE functions) to be able to potentially accommodate the volumes of the already subscribed traffic and of the newly requested subscription, as well as based on appropriate policies reflecting the risks that the provider deems tolerable in satisfying the QoS of the subscribed traffic. At service invocation epochs, admission control determines whether to accept or reject a requested invocation, based on the actual state of the network in gracefully delivering the requested QoS and on fairness criteria amongst the network edges. It is noted that through this approach the potential to efficiently resolve the traditional trade-off between QoS deterioration and network utilisation is increased, since control may be exerted at two levels; the trade-off can be resolved at the expense of rejected subscriptions and/or rejected invocations of accepted subscriptions and/or rejected packets of accepted invocations.

Unlike other schemes, this approach takes actions not only at the call level (determining whether to accept or reject an incoming request), but also at packet level, reducing the rate or downgrading the quality of admitted flows, should conditions (as deduced by the feedback information) warrant so e.g. QoS performance is not delivered. As such, this approach also entails aspects of traffic control at packet level, in addition to the usual service admission control. Further, although it assumes a simple probabilistic criterion for admitting service requests, it could adopt a model-based approach for admitting requests at call level.

Finally, another aspect of this approach is its strong policy-driven nature. It is highly parameterised to allow tuning to the special needs and features of the operational environment. Following its spirit that the risks in asserting QoS deterioration cannot be safely asserted a-priori, it has been designed for "relative parameterisation", based on operational policies following a best-practice paradigm.

10.7.4 Conclusions

Admission control is generally recognized as an essential aspect of any system that aims at providing QoS guarantees. The task that every admission control scheme is requested to undertake is regulating the injected traffic in a way that all existing flows enjoy the QoS they are contracted for, while at the same time trying to maximise the network utilisation.

Solutions for admission control for QoS delivery should scale to fit large networks serving many flows; ideally it should be suitable for the entire Internet. So scalability is a crucial point that any admission control solution must address. Resource reservations per flow using RSVP do not scale because core routers are unable to maintain information for each flow served by the network. Bandwidth Broker (BB)-based solutions imply that admission control decisions are taken at a central location for each administrative domain. In order to cope with scalability, the implementation of the broker should employ mechanisms for distributing the load. Distributed solutions, based on endpoint admission control work per network edge or per host basis, are more scalable.

End-point admission control schemes use model-based or measurement-based approaches to realise their functionality. In both cases QoS deterioration probability is assessed upon service request arrivals. Model-based approaches maintain state information for active services and employ mathematical models, whereas measurement-based approaches rely on either passive or active aggregate measurements. The majority of the admission control approaches utilise information on pre-allocated network capacity, based on which their (model- or measurement-based) logic assesses the current availability of the network and either admits or rejects incoming service requests. It is questionable whether probabilistic QoS guarantees can be safely given at service request times due to the errors inherent in the input required and/or models used for asserting network availability to sustain QoS. The errors in turn are amplified considering the small time period for which the assertion should be valid.

Alternatively, probe-based schemes deduce the ability of the network to sustain the offered load directly, through 'probing' means. These schemes introduce significant latency in response times, and have inherent problems caused by probes stealing bandwidth from established flows and denial of service (when simultaneous attempts congest the network and none is accepted although resources are available.) Other schemes may use off-line functions (traffic engineering) to structure available resources based on user traffic requirements.

The TEQUILA approach [22], tightly couples admission control with the use of traffic engineering (TE) towards a complete solution for QoS provisioning. It utilises the network availability estimates to accommodate QoS traffic, produced by the off-line TE functions, as well as it exploits the adaptability to actual traffic load provided by the dynamic routing and PHB configuration TE functions. The feedback-based model for asserting the risk of QoS deterioration presented by the TEQUILA approach is driven by network state alarms regarding the network's ability to deliver the agreed QoS, rather than trying to predict this network state in short time periods. It is a simple method that ensures minimum service request response times.

The TEQUILA approach in addition to admission control at call level (service request epochs) caters for traffic control at packet level, regulating the volumes of injected QoS traffic of the already admitted flows. Model- or measurement-based approaches refer mainly to call admission control and by their nature mainly focus on trying to avoid QoS deterioration, which if, inevitably, occurs cannot be resolved -it will last for the whole admitted traffic as long as the flows are active.

10.8 Multicast

10.8.1 Introduction

The purpose of multicasting is to provide efficient delivery function for group-based applications that sends the same data to multiple recipients while minimising network load. The key point of multicast is that only one copy of each data packet from the source passes over any link in the network at any time. Copies of the packet are only duplicated when paths diverge at a multicast-capable router, and in this way the bandwidth can be significantly conserved.

10.8.1.1 IP Multicast

The IP multicast service model was proposed by S. Deering, and the relevant specification is presented in RFC 1112 [Deering89]. The basic characteristics of IP multicast are summarised below:

- Each multicast group is identified with a single class D address ranging from 224.0.0.0 to 239.255.255.255 (224/4), and data from any source to a particular group will be received by all the recipients in the group;
- Recipients called multicast group members can be located anywhere in the Internet and can join/leave the group at any time;
- Group members are anonymous to the information source and are locally managed by directly attached routers named Designated Routers (DR) using the Internet Group Management Protocol (IGMP, [Fenner97]);
- Multicast routing protocols are responsible for building distribution trees for delivering data from sources to all group members.

10.8.1.2 Source Specific Multicast

Source Specific Multicast (SSM [Holbro99]) has been proposed as an alternative service model to IP multicast. In SSM, the traditional multicast group is substituted with a multicast channel identified by a tuple (S, G), where S is the IP address of the source and G is the class-D channel destination address. Since each multicast session is on a per-source basis, a unique multicast distribution tree is constructed rooted at the well-known information source. IANA has allocated class D address 232/8 for the exclusive usage of SSM applications. The documentation of SSM deployment has been submitted to IESG for consideration as an Informational RFC in January 2003.

Compared with IP multicast, the SSM model has the following advantages:

- A more scalable usage of class D addresses. In IP multicast, one group session is exclusively identified with a specific class D address, while in SSM, group identification is extended to both source address and class D address. In this case, the whole range of the assigned address space 232/8, specifically for SSM, can be used for each source, resulting in up to 2^{24} available channels per source. Since class D addresses are locally administrated at each particular source, collisions won't take place even if two or more independent senders use exactly the same class D address. This is because (S1, G) and (S2, G) are independent group identifications.
- A more secured service model. As it is mentioned above, IP multicast allows any external source to send data to any group of recipients without any control mechanism. If applications with security requirements need source access control, additional mechanisms could be appended on either routing level or application level. On the other hand, there is always one

distinct source for each SSM group session, and this characteristic automatically prevents irrelevant sources from sending data to group subscribers who are not interested in the data at all.

- A more simple architecture. In SSM, since the address of the unique data source is published with out-of-band mechanisms, group members can make direct subscriptions towards it. This is possible even in the case of inter-domain multicast scenario whereas the traditional IP multicast needs the aid of additional source discovery protocol.
- An easier framework for implementation. Currently the most dominant IP multicast routing protocol is called Protocol Independent Multicast ? Sparse Mode (PIM-SM [Deering94]). The routing level implementation of SSM is known as PIM-SSM [Bhatt00], which is a straightforward adaptation from the PIM-SM protocol by simply eliminating (*, G) group states inside multicast routing entries.

10.8.2 Multicast Group Management

*10.8.2.1 Internet Group Management Protocol (*IGMP*)*

In an IPv4 environment, multicast group management is performed by using Internet Group Management protocol (IGMP) that is run between end hosts and their first hop routers (DR).

In IGMP, the designated router has the responsibility of keeping track of the membership state of the multicast groups that have active members on its sub-network. Each DR periodically sends IGMP packets to check whether the known group members are active or not. If the DR finds active group members, it will activate the underlying multicast routing protocol to join the existing distribution tree. In case there is more than one multicast router on a given sub-network (LAN), one of them is chosen as the DR. After receiving a multicast packet, the router will check if there is at least one member of that group on its sub-network. If there is, the router will forward the message to that sub-network. Otherwise, the multicast packet will be discarded.

IGMP version 2 is designed for the traditional IP multicast service model, and it does not provide the functionality of source filtering. IGMP version 3 is motivated by the advent of SSM. An IGMPv3 packet contains two types of source list, namely INCLUDE and EXCLUDE list. If an IGMP membership report carries a zero-length EXCLUDE list, then the join request is sent as a (*, G) join, otherwise if the report is source specific. In this scenario the DR will issue a corresponding join request towards the sources listed in the INCLUDE list (or not listed in the EXCLUDED list). If the group to be joined is within the SSM address block 232/8, then the DR always send source specific joins. If no source is specified in the group membership report an error message will be issued.

*10.8.2.2 Multicast Listener Discovery (*MLD*) Protocol*

Multicast Listener Discovery (MLD) protocol is designed for multicast group management in an IPv6 environment. Similar to IGMPv3, MLD version 2 (MLDv2 [Vida02]) is designed for Source Specific Multicast in IPv6. A detailed description of MLD is not presented in this document.

10.8.3 Multicast Address Allocation

10.8.3.1 GLOP

As described in Section 10.8.1.1, the IP address block 224/4 is allocated for multicast applications. However the distribution and allocation mechanism of these class D addresses has not met its mature status. GLOP [Mayer00], proposed by IETF, is an experimental mechanism with static allocation of multicast address blocks to individual Autonomous Systems (ASs). IANA has allocated the multicast address block 233/8 for the GLOP usage.

10.8.3.2 The MAAA *Architecture*

Meanwhile, the IETF MALLOC Working Group defines three protocols that work together to form a global dynamic multicast address allocation mechanism. These protocols include:

- A "Host to Address Allocation Server" protocol used by a host to obtain one or more multicast addresses from an address allocation server within its domain.
- An intra-domain server-to-server protocol that address allocation servers within the same domain can use to ensure that they do not give out conflicting addresses.
- An inter-domain protocol know as Multicast Address-Set Claim (MASC [Kummar99]) to provide aggregatable multicast address ranges to domains, which the servers in that domain can then allocate individual multicast addresses out of. This protocol will work in conjunction with another IETF working group IDMR's Border Gateway Multicast Protocol (BGMP [Thaler02]) to provide a scalable inter-domain multicast routing solution.

10.8.4 Multicast Routing Protocols

10.8.4.1 Intra-domain Multicast Routing

10.8.4.1.1 DVMRP/MOSPF/PIM-DM

The Distance Vector Multicast Routing Protocol (DVMRP) defined in RFC 1075 [Waitz88] is a distance vector routing protocol. The basic forwarding mechanism of DVMRP is the Reverse Path Forwarding (RPF) algorithm. RPF is a "flood and prune" algorithm that takes into account group membership to prune those branches of the tree that do not lead to active group members. IGMP is used to detect whether there are group members at the leaves of the tree. This information is passed to routers "up" the tree (i.e. upstream towards the root) in order to prune branches that have no downstream members.

Multicast Extensions to OSPF (MOSPF) defined in RFC 1584 [Moy94] uses the Open Shortest Path First (OSPF) protocol to support multicast routing. In MOSPF group membership reports are flooded throughout the OSPF domain, and MOSPF routers compute source based shortest path tree for each group.

Protocol Independent Multicast Dense Mode (PIM-DM) [Adams02] is very similar to DVMRP. There are two major differences between the two routing protocols: First, DVMRP maintains its own routing table, while PIM-DM directly uses the underlying unicast routing table (constructed by either RIP or OSPF) to perform RPF checks. Second, DVMRP tries to avoid sending unnecessary packets to neighbours who will then generate prune messages based on a failed RPF check. In PIM-DM, multicast routers simply flood packets received on the incoming interface to all outgoing interfaces.

It has been noticed that the above three protocols are all based on flooding algorithms. This characteristic makes it impractical to deploy large scale multicast applications due to their scalability restrictions.

10.8.4.1.2 CBT/PIM-SM/PIM-SSM

Core Based Tree (CBT) defined in RFC 2189 [Ballar97] is the first sparse mode multicast routing protocol standardised by IETF. CBT creates a single shared distribution tree rooted at a unique core node, and all external sources send data to the core while all interested receivers send explicit join requests to the core to become group members. When the core node receives data from sources, it will forward the packets on tree branches created by the join requests from individual group members.

Protocol Independent Multicast Sparse Mode (PIM-SM) [Fenner02], evolved from CBT protocol, is currently the *de facto* multicast routing protocol on the Internet. It has been noticed that though CBT provides a scalable multicast routing solution, the resulted path from sources to individual group members are sub-optimal. In PIM-SM this situation is improved in that group members are allowed to

send join requests along the shortest path back to the source as soon as they have received data coming from the core (known as Rendezvous Point (RP). In this scenario, shortest path trees are created rooting at each data source. In Fig. 1, the receiver R first sends join request to the RP, forming an RP tree branch (Figure 54a). On receiving data from the RP, R knows about the address of the source S, and then it will send a shortest path tree (SPT) join directly to it. Once R receives data via the shortest path, it will prune itself from the original RP tree to avoid traffic loop (Figure 54b).

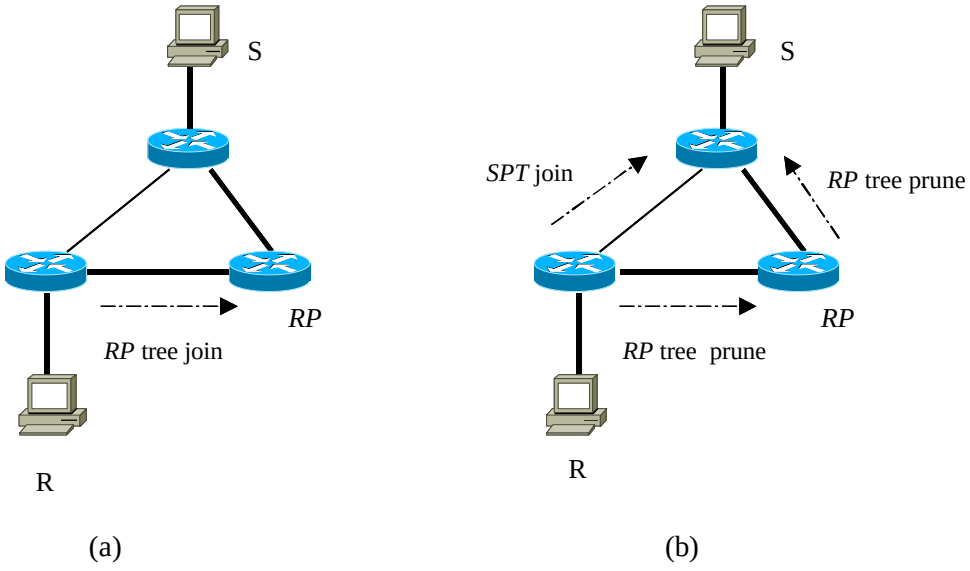


Figure 54 *PIM-SM* routing protocol

PIM-SSM [Bhatt00], recently developed for Source Specific Multicast, is an adapted routing protocol to PIM-SM. In PIM-SSM, group members obtain source addresses by out-of-band mechanisms, and hence they are able to include the source list in the IGMPv3 group membership report. In this case the designated router (DR) will send direct join requests to individual sources. It should be noted that there is no RP in the PIM-SSM distribution tree, and on tree routers don't maintain any (*, G) group state as they do in PIM-SM.

10.8.4.2 Inter-domain Multicast Routing

Inter-domain multicast routing has evolved from the need to provide scalable, hierarchical and Internet-wide multicast services. It has been realised that intra-domain protocols cannot fulfil many tasks such as source discovery in foreign domains, global multicast address allocation etc. IETF has proposed both near-term and long-term solutions for deploying global multicast services on the Internet.

10.8.4.2.1 MSDP/MBGP

Being an intra-domain multicast routing protocol, PIM-SM only constructs a distribution tree within a single domain. Given the fact that multicast sources and group members may located anywhere in the Internet, one of the problems is how to inform an RP in one domain about active sources in foreign domains. The near-term solution to this problem is Multicast Source Discovery Protocol (MSDP) [Farina98]. In this protocol, representatives known as MSDP peers have the task of announcing local active sources to their counterparts in other domains. Once an MSDP peer learns of any active local sources, it will send Source Active (SA) message to all its directly-connected MSDP peers. This type of SA flooding is based on inter-domain-wide RPF checking. Once the RP of the foreign domain learns about the address remote source, it will send an inter-domain join request to the source. In this scenario, a global multicast distribution tree is constructed.

Multi-protocol Border Gateway Protocol (MBGP [Bates98]) is an extension to BGP4, and it is able to carry multi-protocol routes by adding the Subsequent Address Family Identifier (SAFI) to two BGP4 messages: MP_REACH_NLRI and MP_UNREACH_NLRI. MBGP is used in multicast routing when

an inter-domain group join message is sent from an RP towards remote sources in foreign domains. It should be noted that MBGP only carries path information of sources but does not contain any multicast group information. The joint working mechanism of MSDP and MBGP provides a complete solution for global deployment of multicast services.

As far as Source Specific Multicast (SSM) is concerned, this service model does not need source discovery mechanism such as MSDP, however MGBP is still necessary for explicit group join towards sources in foreign domains.

10.8.4.2.2 BGMP

The Border Gateway Multicast Protocol (BGMP [Thaler02]) is the first long-term solution to Internet-wide multicast development. BGMP, working together with MASC, tries to provide a framework that copes with both inter-domain multicast routing and class D address allocation. The key idea of BGMP is to construct a bi-directional shared tree across multiple domains using a single root. The “root domain” is decided by a strict class D address allocation scheme using MASC. As it is mentioned in section 3.2, MASC assigns specific multicast address ranges to individual domains. This strategy is based on the assumption that if a particular domain owns the address for a particular group, this domain will be significantly involved in the multicast session. As a result, BGMP will select this domain as the root of the multicast distribution tree that might cross multiple domains.

10.8.5 IP multicast's current business practices

10.8.5.1 Deployment of the multicast technology

The following chart provides a snapshot of the most recent IP multicast activity that has been processed by a subset of IP multicast-capable routers deployed in the Internet, courtesy of CAIDA (www.caida.org/tools/mantra). This chart provides on a monthly basis a snapshot of the statistical information related to the number of multicast sessions and the corresponding group members, according to the monitoring of all the routers that participate in the maintenance of the associated multicast route entries.

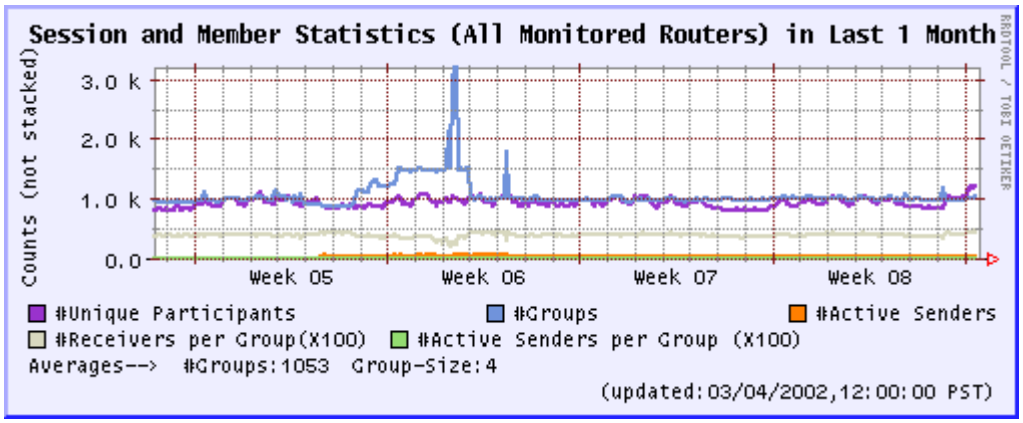


Figure 55: session and member statistics on a monthly basis

This graph shows there is an average number of more than 1000 multicast groups, with an average number of around 400 receivers per group.

A multicast host can either be a sender, a receiver or both. A host is said to be "participant" if it is a member of at least one multicast group. The total number of multicast participants indicates the total number of (S, G) pair entries that are maintained by the routers. From this standpoint, if a host is a member of n groups, it will be counted n times.

This chart shows that IP multicast is *currently deployed and operational*, while there are around 15 service providers worldwide who claim to provide an IP multicast service, even though such a service

simply consists in establishing a tunnel towards the M-Bone, so as to allow the retrieval of IETF conferences, for example.

10.8.5.2 Multicast-based services

IP multicast technology cannot really be sold as a service of its own, but rather as a means to provide value-added IP services that would gracefully benefit from such a technology. Therefore multicast connectivity to the multicast enabled Internet is mostly an included part of the basic service offering for no extra charge. The value-added IP services address both the residential and the corporate market. A non-exhaustive list of such services is provided below:

- TV and radio broadcasting.
- Live broadcasting (e.g. concerts, conferences and training courses).
- Video on demand.
- Network games.
- Videoconferencing.

The M-Bone has been available since the beginning of the 1990s and broadcasts a large amount of (but not exclusively) university-related content in multicast. Radio stations, TV channels and concerts are also broadcast on MBone.

10.8.5.3 Business model

The following picture depicts a proposal business model that could be applied to services run over multicast. We assume ISP (Internet Service Provider)'s network, Transport Operators networks and Content Provider network if any, are all multicast enabled.

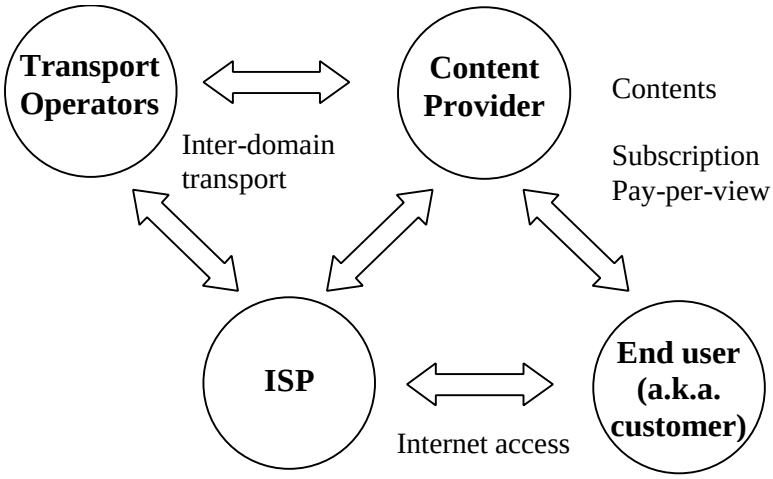


Figure 56: business model

A Content Provider owns and sells contents such as music and video. This content are retrieved by streaming (e.g. radio program) or by file downloading (e.g. CD burning).

An ISP physically connects a Customer to the Internet. If a Customer and his Content Provider are not connected to the same ISP, some Transport Operators are involved to relay the data throughout the Internet.

A Content Provider charges his Customers by subscription (e.g. per month) or pay-per-view (e.g. per movie). An ISP can charge directly his users for the service. The ISP will in return give back part of the fee to the Content Provider. In this scheme a customer sees her ISP as a Content Provider.

10.8.6 QoS in IP multicast

10.8.6.1 Multicast and DiffServ

QoS in IP multicast delivery has not so far received significant interest in the IETF, even in DiffServ which has been the most active QoS working group for many years.

10.8.6.2 Multicast and PHB

By essence the DiffServ PHB paradigm is multicast compliant. A router applying a PHB to a datagram does not care whether it is unicast or multicast routed. Whenever a datagram enters a router, the router selects the PHB according to the DS field value (DSCP), in no way (in the pure DiffServ approach, and ignoring boundary routers) it would take into account other pieces of information like destination address, application port, physical interface and so forth.

10.8.6.3 Multicast and resource provisioning

In general, the more one knows about the paths followed by data flows, the better one should be able to provision the network. The IP multicast paradigm (at least the original one [Dee01]) allows any user to send or receive to/from a group. This feature makes it possible for branches to be easily and dynamically grafted to the distribution tree or pruned. However, a multicast flow is not always that much less predictable than a unicast flow. For example the set of destinations to which unicast packets are sent in a connection to the global Internet, is larger and more dynamic than the set of destinations to which multicast packets are sent among VPN sites.

10.8.6.4 Multicast and SLS

DiffServ is basically a sender-oriented mechanism. If you want to allow SLS with receiver-oriented requirements you must provide a mean to convey the requirements up to the source to mark effectively the traffic, as early as possible, with the proper DSCP. The problem does already exist for unicast, but is of course another order of magnitude higher for multicast. More precisely the specificity of multicast is twofold: first, different receivers can require different service levels; second, new receivers can join dynamically an already existing distribution tree.

10.8.6.5 Standardisation efforts

No RFCs and no working group drafts have been published on QoS multicast. The only significant contribution to the theme is an individual draft [Bless02]. This draft has never reached the status of working group draft nor has been approved as RFC. Therefore, as far as the IETF standardisation process is concerned, we can hardly call it an IETF standardisation effort. At most, we can say that this effort has been unsuccessful. We provide in the next paragraph the main ideas of this document.

10.8.6.5.1 Draft-bless-DiffServ-multicast-05.txt

The core contents of this draft deals with the so-called Neglected Reservation Sub-tree Problem.

10.8.6.5.1.1 Neglected Reservation Sub-tree Problem (NRS Problem)

A DiffServ capable node would copy the content of the DS field into the IP packet header of every replicate. Consequently, replicated packets get exactly the same DS code-point (DSCP) as the original packet, and, therefore experience the same forwarding treatment as the incoming packets of this multicast group.

When a new receiver joins an IP Multicast group, the multicast tree is expanded by a new branch, connecting the new receiver to the already existing multicast tree. As a result of tree expansion, the new receiver will implicitly use the improved QoS of the tree, because of the copied "better" DSCP.

If the additional amount of resources that is consumed by the new part of the multicast tree is not taken into account by the domain management (cf. section 1.1), the currently provided level of quality of service of other receivers (with correct reservations) will be affected adversely or even violated.

10.8.6.5.1.2 Solution for the NRS Problem

A new receiver joins a multicast group that is using a DiffServ service. Multicast routing protocols achieve the connection of the new branch to the (possibly already existing) multicast delivery tree as usual.

The unauthorised use of resources is avoided by re-marking at branching nodes all additional packets leaving downwards the new branch. At first, the new receiver will get all packets of the multicast group without any guaranteed quality of service (i.e. best-effort only).

If a pre-issued reservation is available for the new branch or an entity (receiver, sender or a third party) issues one, the management entity instructs the branching router to set the corresponding code-point for the demanded service.

10.8.6.6 QOS-aware multicast routing algorithms

10.8.6.6.1 The Steiner Tree Problem

The problem of plain multicast routing can be modelled as finding optimal solution to the Steiner tree problem, which has been proved to be *NP-Complete*. A network can be defined as a bi-directional graph $G = (V, E)$ with node set $V(|V| = n)$ and link set *L*. Each link in the graph is attached with a metric of link cost. There also exists a source node *s* and a receiver set $R \subset V$. The Steiner tree problem is to minimise the total cost of tree *T* rooted at s and spanned to all the nodes in *R*, i.e.,

$$\text{Minimise} \sum_{(i,j)\in E} C_{ij} Y_{ij} \quad i, j \in V$$

$$\text{where } Y_{ij} = \begin{cases} 1 & if\ (i,j) \in T \\ 0 & otherwise \end{cases}$$

The *KMB* heuristic [Kou81], proposed by Kou, Markowski and Berman, applies Prim's minimum spanning tree algorithm to the complete-distance graph *G*', where *G*' is a graph that contains all node in *R*+{*s*}, and the cost of each link in *G*' is the shortest distance between the two pair of nodes in *G*. The time complexity of *KMB* is $O(mn^2)$ where $m = |R|$.

In *TM* heuristic [Taka80], the first step is to compute the shortest distance from all nodes in *V* to each member in *R*. When the Steiner tree is constructed, the group member that has the shortest distance to partially built Steiner tree is selected to join using a greedy approach. When all the members in *R* have been included, the *TM* heuristic terminates. The time complexity of *TM* is also $O(mn^2)$.

10.8.6.6.2 *QoS* Constrained Steiner Tree Problems

The Steiner tree problem has been extended to include additional constrained *QoS* metrics, such as delay, delay variation and bandwidth capacity etc or even the combination of multiple constraints. Since all this constrained problems are extensions to the Steiner tree problem, all of them are *NP-Complete* as well. Table 10 presents a summary of relevant heuristic solutions to these problems.

Name	Constraints	Type	Time complexity
KPP [KPP93]	Delay (bound by Δ)	Centralised	$O(\Delta n^3)$
BSMA [Zhu95]	Delay	Centralised	$O(kn^3 \log(n))$ [1]
Kompella [Kompe93]	Delay	Distributed	*N/A*

Jia [Jia98]	Delay	Distributed	*N/A*
CCDVMA [Rousa97]	Delay + Delay variation	Centralised	$O(mn^4)$
Jia [Jia97]	Bandwidth	Centralised	$O(m^3n^2)$
GTM [Low00]	Bandwidth	Centralised	$O(m^3n^2)$
Low [Low02]	Delay + Bandwidth	Centralised	$O(TBm^2n)$ [2]

1. *k* is the parameter in tree-switching using *k-shortest path*
2. $TB = \sum_{t=1}^{m} B_t$ where B_t is the bandwidth requirement (in unit) of tree *t*.

Table 10 *QoS* constrained Steiner tree heuristics

10.8.6.7 QoS-aware Multicast Routing Protocols

10.8.6.7.1 Yet Another Multicast (*YAM*)

Yet Another Multicast (YAM [Carlb97]) proposes a scalable approach for building shared trees, by providing multiple routes from the receiver to the existing tree with one-to-many joining mechanism. QoS routing in the context of YAM is achieved by discovering multiple paths from a receiver to an existing tree and selecting the one that satisfies certain QoS requirements e.g. link capacity, reliability. YAM, like PIM-SM, operates independently of any underlying unicast protocol. It also deals with the asymmetric links in the network.

In implementation, the receiver restricts its multi-path spanning join by performing a bid-order broadcast with limited scope of Time-To-Live (TTL) field. On tree routers that receive the broadcast message become candidate routers and return a bid message containing QoS path information to the potential group member. Finally, this new receiver examines all the bid messages and selects a proper path that satisfies the required QoS to join the distribution tree.

10.8.6.7.2 QoSMIC

QoSMIC [Falou98] is a multicast routing protocol for supporting QoS-sensitive multicast applications and can been considered as an extension of YAM. There are two join mechanisms provided by QoSMIC, namely local search and tree search.

In the case of local search procedure, a similar procedure as in YAM is used. On the other hand, there exists a manager router that has the responsibility of handling new member joins. When the joining router initiates the local search, it also contacts the manager router. If the manager router has sufficient knowledge of tree structure and the network topology, it sends “bid-order messages” to candidate on-tree routers; otherwise, it multicasts a “bid-order message” on the tree. Afterwards, the candidate routers unicast “bid messages” towards the joining router.

Finally, the new router will select the “best path” among the candidate ones based either on the static QoS metrics or on the dynamic routing information collected by the “bid messages” in order to satisfy the needs of the applications. In the following, the new router sends a join message across the path chosen to graft to the existing tree. The local and multicast tree searches are illustrated in Figure 57.

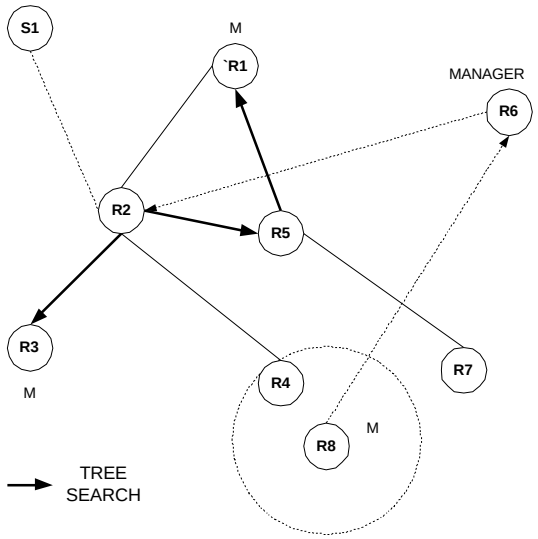


Figure 57 *QoSMIC* routing

10.8.6.7.3 QoS-aware Multicast Routing Protocol (QMRP)

QoS Multicast Routing Protocol (QMRP [Chen00]) starts the search of a feasible path to a delivery tree with a single path and expands to multiple paths when it is necessary according to network conditions. More specifically, when a new receiver wants to join a group, it first has to inquire the Session Directory about the address of the core of the multicast tree. Then, it sends a *REQUEST* message to the core following the unicast routing path. This *REQUEST* message carries the QoS requirements of the application (e.g. bandwidth) and proceeds to the unicast routing path as long as the intermediate nodes meet the requirements for the resources needed. If every intermediate node can satisfy the QoS demand, a feasible path is found to the delivery tree by traversing only a single path, similarly to PIM-SM protocol. Otherwise, if an intermediate node does not have the required resources, the discovery is expanded to multiple path routing searches. In this case, a *NACK* message is sent from this node to the previous one, which in turn sends *REQUEST* messages to all neighbour nodes except the ones from which *REQUEST* and *NACK* are previously received. From this point and on, each *REQUEST* message will try to search a feasible sub-path to the delivery tree. Once the path reaches an on-tree node, an *ACK* message is sent towards the new member. If the new member receives more than one *ACK* message, it selects the "best path" to the delivery tree and rejects the other ones.

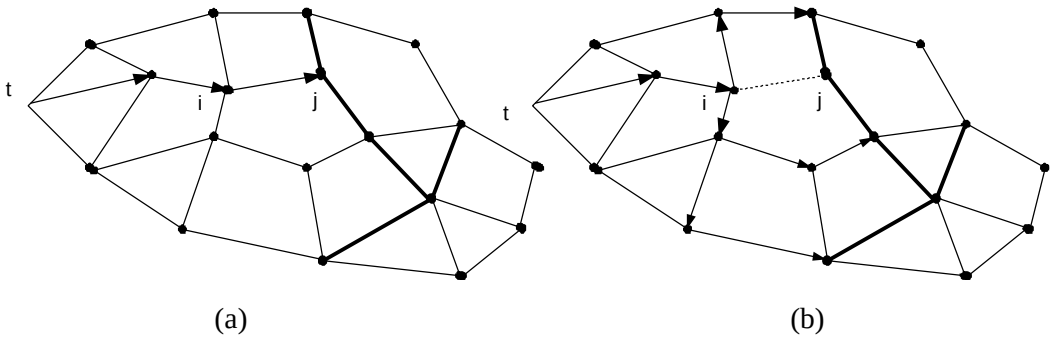


Figure 58 *QMRP*

An example of the above procedure id depicted in Figure 58. In Figure 58a, all the intermediate nodes have the required resources and a single feasible path is found between the new member and the tree

(bold links), while in Figure 58b node *j* has not the necessary resources and a multiple path search is initiated.

10.8.6.7.4 Multicast *QoS* (*MQ*)

The MQ protocol [Yang01] was proposed to support multimedia group communications with QoS guarantees for heterogeneous recipients.

Being an integrated solution, MQ sets up a multicast distribution tree with quantitative QoS requirements, and makes explicit bandwidth reservation for each group member during the phase of tree construction. When there exist heterogeneous receivers, resources are reserved up to the point where the paths to different receivers diverge. From the receiver's point of view, MQ merges the join requests with heterogeneous QoS demands from different receivers at the point where multiple requests converge. When a join request propagates upstream towards the source, it stops at the point where there is already an existing QoS reservation that is equal to or greater than that being requested. Figure 59 basically illustrates how different resource reservations are merged along the multicast join procedure. Suppose the requests from receiver A, B and C demands 10Mbps, 512kbps and 56kbps bandwidth respectively, their reservations are merged to the highest request at each hop as shown in the figure. MQ can also adapt to resource consumption with dynamic group membership. For example, if an on-tree router detects that the departing receiver originally requested the highest QoS, it will automatically shrink its reservation or even reshape the distribution tree to exactly satisfy the remaining participants. In Figure 59(b), we can find that when receiver A with the bandwidth requirement of 10Mbps wants to leave the multicast session, the remaining receiver B with 512kbps requirement will switch from the original "shared" path (S→R1→R2→R4) with the capacity of 10Mbps to a shorter one (S→R3→R4) which still satisfies its QoS demand for bandwidth optimisation purpose.

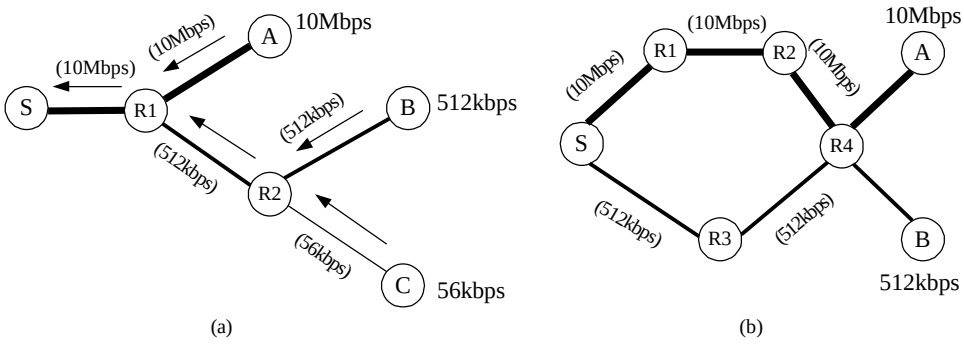


Figure 59 *MQ* tree dynamics

10.8.6.8 Multicast in Differentiated Services

Due to the emergence of various group communications with heterogeneous QoS requirements from individual group members, some research effort has recently been focused on multicast deployment in Differentiated Services (DiffServ [Blake98]) networks. It has been deemed that the DiffServ architecture is a promising solution to achieve service differentiation in a large scale, thanks to its scalable router implementation in the core network. Related research works on multicast in DiffServ domains are summarised in the rest of the chapter.

10.8.6.8.1 Neglected Reserved Sub-tree (*NRS*)

B. Bless et al first pointed out one fundamental problem known as "Neglected Reserved Sub-tree (*NRS*)" in multicast deployment inside DiffServ networks [Bless02], and described in Section 10.8.6.5.1. The authors found that in DiffServ domains network resources are consumed based on the pre-negotiated Service Level Agreement (*SLA*). However, in a DiffServ-aware multicast environment, it is possible that the actual resource consumed exceed the pre-negotiated *SLA*. Since the multicast tree

could branch at any core router, the amount of outgoing traffic from a domain may exceed the incoming traffic rate to the domain and thus consume additional resources. This scenario is illustrated in Figure 60, where router *R2* joins group without traffic conditioning at the edge of the DiffServ domain. It can be noticed that if the *ISP* allows inter-class bandwidth pre-emption, the over-reservation of higher class of service can also influence the performance of lower class of services.

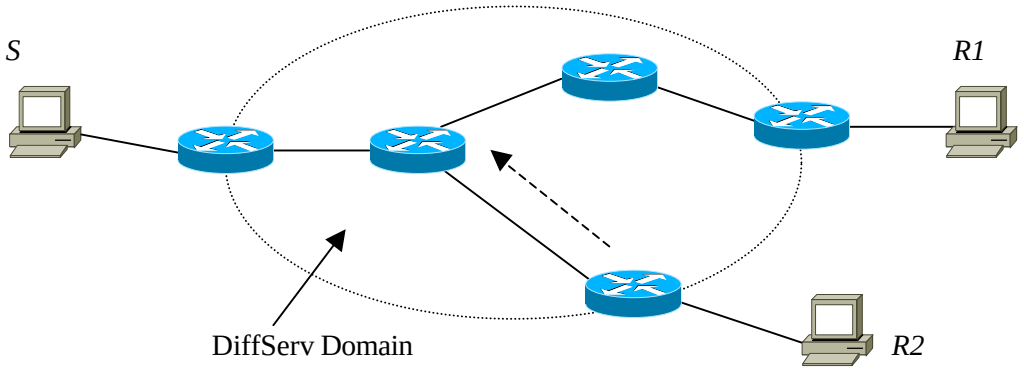


Figure 60 The *NRS* Problem

The NRS problem can be solved by assigning a Lower than Best Effort (LBE) PHB to the newly branched traffic. In the approach, the resources and processing of existing traffic are protected while maintaining the simplicity of the DiffServ model. In order to obtain higher level of services, the joining node has to explicitly negotiate with the Bandwidth Broker (BB) for resource reservation. In case that the BB can allocate available bandwidth to the new branch, the new group member will receive the traffic based on its originally desired QoS class. Otherwise this branch has to remain in the LBE service.

10.8.6.8.2 DiffServ Multicast (*DSMCast*)

DSMCast [Strieg01] is a scalable framework that tries to achieve complete stateless multicast in DiffServ networks. The main idea of DSMCast is that both the destination address of individual receivers and their QoS requests are embedded in the head of group data packets instead of being maintained inside of the DiffServ domain. Packets are duplicated where necessary at core routers and delivered to individual receivers based on their unicast destination address contained in the packet head. In this sense, DSMCast does not make used of class D address in the traditional IP multicast service model. During the duplication procedure, DSCP value is also remarked according to the QoS requirements of individual downstream group members. In this scenario, core routers need maintain neither QoS states nor multicast group states, and this characteristic guarantees the high scalability requirement of the architecture.

On the other hand, DSMCast aborts the traditional IP multicast service model that has already been very popular throughout the Internet. Moreover, in case of large number of egress routers or receivers, DSMCast data transmission becomes inefficient due to the relatively longer packet head that contains individual receiver's address and its desired QoS class.

10.8.6.8.3 QUASIMODO

QUASIMODO [Bianch03] is a complete DiffServ multicast framework based on the IP multicast service model. The objective is to (1) provide flexible QoS support with respect to heterogeneous multicast groups, and (2) maintain compatibility with currently deployed multicast protocols.

In QUASIMODO, PIM-SM is selected as the reference multicast routing protocol. In order to accommodate QoS heterogeneity, DiffServ extensions have been made on existing PIM-SM join requests and multicast forwarding table inside core routers. First, if a potential member decides to join

the group with a certain level of QoS class, it will send out an adapted IGMP report (*, G, q) where q indicates the DiffServ service class this receiver desires to receive. Once the Designated Router (DR) receives the report, it will issue a (*, G, q) join request towards the RP, and this join request will explore a new tree branch that satisfies the demanded QoS class. On the other hand, in order to handle join requests with heterogeneous QoS demand, the multicast forwarding table inside core routers also needs to be extended accordingly. Specifically, the outgoing interface (oif) field of each group is appended with an additional DSCP entry, which is used to mark replicated packets that are forwarded on this particular outgoing interface. Table 11 presents a typical structure of a DiffServ aware multicast forwarding table in QUASIMODO.

Group address	*iif*	*oif*	*DSCP*
226.214.18.5	*A*	*B*	*AF11*
		C	*AF21*
235.66.123.16	*D*	*A*	*AF31*
		C	*AF41*

Table 11 *QUASIMODO* multicast forwarding table

The routing dynamics of a particular group in QUASIMODO is basically how to update oif list as well as its associated DSCP filed in the multicast forwarding table according to the received join requests with various QoS requests. There are basically three cases when a core router receives a group G join request:

- The interface is not in the oif list of G. In this case the router will include this interface into the oif list, and record the desired QoS class carried in the join request to the DSCP field of the forwarding entry. If this core router is not included in the distribution tree, it will forward the join request towards the RP or sources.
- The interface is in the oif list and has higher QoS class state than that is indicated in the join request. In this case the core router need do nothing.
- The interface is in the oif list of G but has lower QoS class state than the one indicated in the join request. In this case the core router will upgrade the DSCP value associate with this oif with the one that is carried in the newly arrived join request. Meanwhile a new join request with higher QoS class will be sent towards the RP or source, so that the QoS requirement of the new downstream member can be satisfied.

When group data is received on the iif, the core router will duplicate the packet and forward its copies on all its interfaces in the oif list. The forwarding behaviour on each outgoing interface is uniquely based on the corresponding DSCP field in the group forwarding entry.

10.9 ipV6

10.9.1 Review of IPv6 QoS *à la DiffServ*

The definition of QoS *à la DiffServ* has been integrated in the specification of the IPv6 protocol right from the beginning. Indeed, in the IPv6 RFC [RFC-2460], the 8-bits field called "Traffic Class" has been described to allow services differentiation as defined in [RFC2474]. This field commonly known as the DiffServ (DS) byte, is composed of two parts:

- The first six bits of this field are used as a code-point, the DiffServ Code Point (DSCP), to select the PHB (Per Hop Behaviour) that a packet experiences at each node.
- The two remaining bits are currently unused (CU) and reserved for routers in the case where there is a congestion risk, in combination with the RED (Random Early Detection) algorithm. Differentiated-services compliant nodes, when determining the per-hop behaviour to apply to a received packet, ignore the value of these CU bits.

The Traffic Class field structure is presented below:

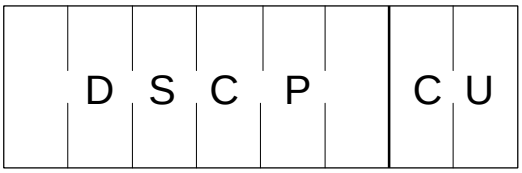


Figure 61 Traffic Class byte format

In summary, the 8-bit Traffic Class field in the IPv6 header is available for use by originating nodes and/or forwarding routers to identify and distinguish between different classes or priorities of IPv6 packets. The Traffic Class field in the IPv6 header is intended to allow similar functionality to be supported in IPv6 as in IPv4 DS field bits.

The following general requirements apply to the Traffic Class field:

- The IPv6 service within a node must provide a means for an upper-layer protocol to supply the value of the Traffic Class bits in packets originated by that upper-layer protocol. The default value must be zero for all 8 bits.
- Nodes that support a specific (experimental or eventual standard) use of some or all of the Traffic Class bits are permitted to change the value of those bits in packets that they originate, forward or receive, as required for that specific use. Nodes should ignore and leave unchanged any bit of the Traffic Class field for which they do not support a specific use.
- An upper-layer protocol must not assume that the value of the Traffic Class bits in a received packet is the same as the value sent by the packet's source.

The complete definition and usage of the Traffic Class field is described in [RFC2474].

10.9.2 Flow Label exploitation

The Flow Label is a 20-bit field included in every IPv6 packet header. The current IETF Flow Label specification document is [DRAFT-FLOWLABEL]. The previous version of the specification included too many disparate details about the way the Flow Label should or could be used. It was too complicated, so it needed to be rewritten. Version 4 of the document specifies only the Flow Label field, requirements for IPv6 source nodes labelling flows, and requirements for flow state establishment methods.

Packets are labelled by the source to identify a flow. The value has no mathematical or any other meaning. Unlike the Traffic Class field, the Flow Label value fixed by the source must be delivered unchanged to the destination. An intermediate router can use this value to apply a specific treatment to the packets. A flow is identified by the triple (source address, destination address, Flow Label). To enable flow-specific treatment, flow state needs to be established on the path from the source to the destination. There is no proposed solution in the specification; the only two requirements are that a solution must provide a state clean-up mean, and a method to recover from the case where the requested state can't be supported.

One possibility offered by the Flow Label could be to use it as an extended DSCP field thanks to its 20-bit length. But the fact that the value must not be modified on the path from the source to the destination may be a constraint. Nevertheless, up to now, there is no concrete proposal on how a flow label path should be created. The only proposition, which has expired, has been tackled in [DRAFT-QOS-FLOW].

10.9.3 Possibilities offered by extension headers

In IPv6, optional internet-layer information is encoded in separate headers that may be placed between the IPv6 header and the upper-layer header in a packet. There are a small number of such extension

headers, each identified by a distinct Next Header value. New headers can be defined in order to implement a new service or option, without modifying the core IPv6 protocol specification.

Regarding inter-domain QoS, some information exchange or other mechanism could take advantage of this feature offered by IPv6. Indeed, the technical solutions responding to the needs of defined processes or mechanisms could be built on a new IPv6 header. For instance, it may be possible that one interesting solution consists in making some kind of source routing, based on the information of the nodes and/or the ASs and/or the CoS to go through successively. This would ensure that the required QoS needs are met along the defined path to the destination. This could rely on the definition of a new header or the completion of the Routing header already defined.

The Routing header is used by an IPv6 source to list one or more intermediate nodes to be "visited" on the way to a packet's destination. This function is very similar to IPv4's Loose Source and Record Route option. The Routing header is identified by a Next Header value of 43 in the immediately preceding header. Among others, the Routing header contains two fields that could be exploitable:

- Routing Type: 8-bit identifier of a particular Routing header variant. At this moment, only Routing type 0 and its usage has been defined.
- Type-specific data: Variable-length field, of format determined by the Routing Type, and of length such that the complete Routing header is an integer multiple of 8 octets long. For Routing Type 0, this data is a list of IPv6 addresses for the nodes to go through to the destination.

This offers the possibility of defining a new Routing type, with the corresponding specific data to be handled. For instance, it could contain the list of the ASBR (Autonomous System Border Router) to go through along the path, possibly with the QoS classes to apply in the corresponding ASs thus forcing a path to reach the destination with the required QoS. One advantage is that the path followed (both from the crossed ASs and the applied QoS point of view) will be kept.

A new header could also be specified (instead of using a specific type of Routing header) in this aim. It would contain the ASs and the QoS classes to be applied along the path. What's worth laying stress on is that the information contained in these headers would only be treated by ASBR. This wouldn't have any impact on intra-domain routers.

Of course, this is only one marginal example of what could be done with IPv6 extension header. It is only given to illustrate that IPv6 brings the advantage of allowing the definition of new headers the mechanisms imagined for inter-domain QoS could rely on. Up to now, no real work or specification neither proposition has been submitted to the IETF community that would cover this area.

10.9.4 MBGP considerations

The inter-domain routing protocol used for IPv6 is MBGP [DRAFT-MBGP]. Only two new attributes need to be defined: Multiprotocol Reachable NLRI (MP_REACH_NLRI) and Multiprotocol Unreachable NLRI (MP_UNREACH_NLRI). Those attributes aim at advertising IPv6 address format. They are optional and non-transitive, which means that a router that does not implement MBGP has to ignore these attributes and must not forward them.

One proposal to exchange QoS attributes between domains is specified in [Cri01]. This specification defines a new attribute (QOS_NLRI), as described in *S*ection 10.4.2.3. This attribute is optional and transitive, and is made for IPv4 only (the Next Hop Address field is 4 byte long).

The difference about the transitiveness means that:

- A domain that does not implement MBGP will not forward IPv6 (and so IPv6/QoS) prefixes.
- A domain that implements MBGP but does not implement QOS_NLRI attribute will forward IPv6/QoS prefixes even though it does not take into account this QoS within its own domain.

A solution could be to merge attributes of QOS_NLRI and MP_REACH_NLRI, and to create a new attribute that provides us with an IPv6/QoS prefix. Such an attribute could be named MP_QOS_NLRI. It would be defined as follows:

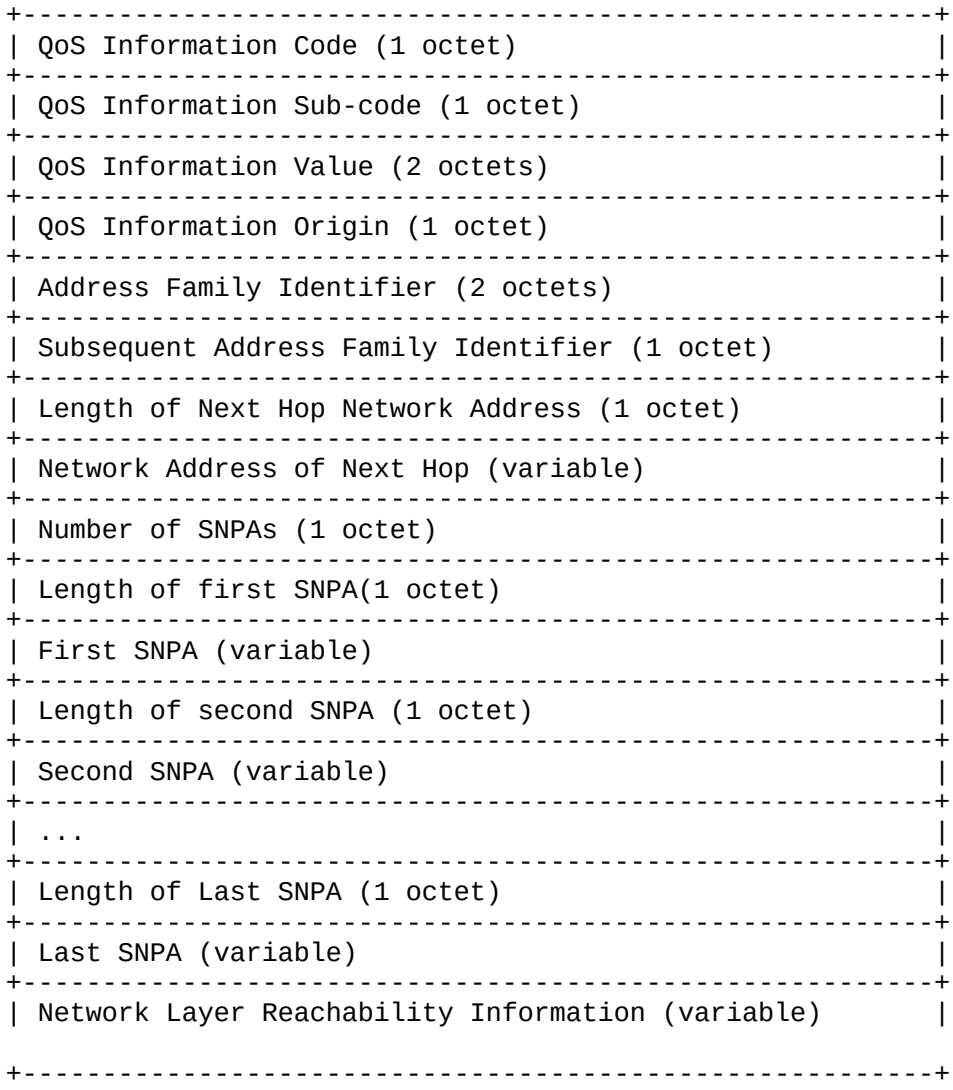


Figure 62 MP_QOS_NLRI attribute

The first 4 fields ("QoS Information…") are taken from the specification [DRAFT-MBGP] and are used to code the QoS information. Other fields are from the MBGP specification [Cri01].

In order for BGP speakers to agree on their IPv6/QOS capabilities, they must use the Capability Advertisement procedures as defined in [RFC2842].

10.10 Policy-based Networking

10.10.1 Introduction

Policy-based networking has attracted a lot of attention in recent years, seen as a flexible approach for the coordinated configuration of network devices through high-level directives translated to the required low-level actions. Policies have been studied in the research community almost for a decade as a means for implementing flexible and adaptive systems for management of internet services, distributed systems and security systems. These are typically large-scale systems, which require management solution which are both self-adapting and that dynamically change the behaviour of the managed system. Policies are defined as rules that govern the choices in behaviour of a system [Slom94]. There are still research issues with analysing policies for conflict detection and refinement in order to build policy-based management systems. After the evolution of Quality of Service models in IP networks, the IETF has been investigating policies as a means for managing IP-based multi-

service networks, focusing more on the specification of protocols (e.g. COPS) and the object-oriented information models for representing policies.

In the this section, we will present the state of the art in the area of policy-based management starting from a description of the architectural components of policy-based frameworks defined in the literature and in IETF Working Groups. We will then review the high-level policy languages defined in order for the administrator to describe the policies, the information models defined by IETF for representing and storing policies, technologies used for enforcing the policies entered by the administrator, and the policy protocols defined in the IETF for conveying policy-related information. Finally, we will present some of the commercial policy-based management products.

10.10.2 Policy Frameworks

10.10.2.1 Policies in the Research Community

A lot of research has been carried out in the area of policies for the management of distributed systems, with most of the concepts pioneered by Imperial College London focusing mostly in the specification of policy definition language described in section 3. Policies are specified as objects, which define a relationship between *subjects* (managers) and *targets* (managed objects). Policies are separated from the automated managers, facilitating the dynamic change of the behaviour and the adaptivity to new requirements without re-implementing the management applications. Domains provide the framework for partitioning management responsibilities by grouping objects in order to specify a management policy that applies to a domain. The following types of policies (modalities) are identified in [Damian01]: **Authorisation Policies** (positive and negative), which specify what a subject is authorised/forbidden to do with respect to a set of managed objects. These are essentially access control policies. **Obligation Policies**, which specify what operations the subject *must* perform on a set of target objects. Positive obligation policies are triggered by events. **Refrain Policies**, which define the actions that subjects must not perform on target objects. **Delegation Policies** (positive and negative), which specify which actions subjects are allowed to delegate to others. [Slom99] gives a general description of the policy framework depicted in Figure 63. An administrator creates and modifies policies using the Policy Editor. Authorization policies are disseminated to target agents as specified by the target domains and obligation policies to manager-agent applications as specified by the subject domains. Manager agents interpret policies, which can be enabled, disabled or removed from the application and register with the monitoring service to receive events that trigger one or more policies. On receiving an event, the manager-agent queries the domain service to determine the target objects, and performs the policy actions on them. Detailed description of the components of this architecture as well as the interactions between them is provided by [Marr96].

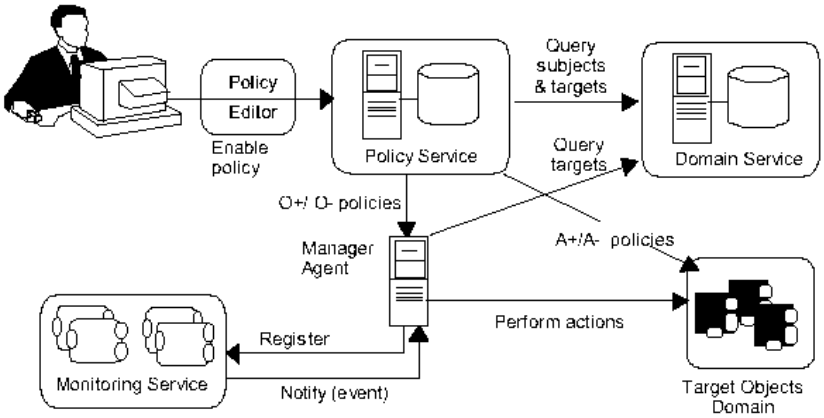


Figure 63 Policy Management Architecture [Slom99]

10.10.2.2 Policies in IETF

Two working groups in the IETF have considered policy management or *policy-based networking*: the Resource Allocation Protocol (RAP) Working Group (WG) [RAPWG] and the Policy Framework WG [PolicyWG]. The latter has produced a draft describing a general framework for representing, managing, sharing and reusing policies in a vendor independent, interoperable and scalable manner but they decided to withdraw the work on it in order to be consistent with framework document produced by the RAP WG. The RAP WG has described a framework for policy-based admission control specifying the main architectural elements [RFC2573]: the Policy Management Tool (PMT), the Policy Repository, the Policy Enforcement Point (PEP) and the Policy Decision Point (PDP).

PMT enables an entity to define, update and optionally monitor the deployment of Policy Rules, while the Policy Repository is responsible for storing policy rules in a structured format providing also the ability to the other components to retrieve them. PEP represents the component that always runs on the policy-aware node and it is the point where the policy decisions are actually enforced. The PDP is responsible for retrieving the policy rules from the repository and it is the point where the policy decisions are made. The above framework is described in Figure 64 as well as the interactions between the components.

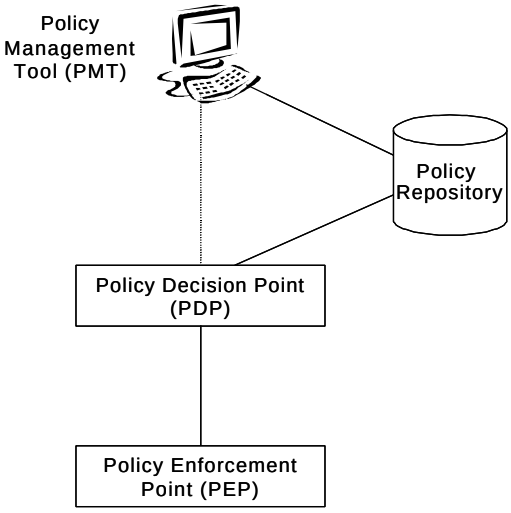


Figure 64 IETF Framework

The framework supports two models: the outsourcing and the configuration model. In the outsourcing scenario, the PEP delegates responsibility to an external policy server (PDP) to make decisions on its behalf while in the configuration model the PDP may proactively provision the PEP reacting to external events (such as user input), PEP events, and any combination.

10.10.3 Policy Specifications

Most of the work that has been done in the research community deals with the specification of policies. Although IETF produced an initial draft describing a policy framework definition language they decided to stop the work continue only the work representing policies according to an in Information model presented in the next section. The Ponder language for specifying Management and Security policies [Damian01] evolved out of work on policy management at Imperial College over a period of about 10 years. Ponder is a declarative, object-oriented language that can be used to specify both security and management policies. It supports obligation policies that are event triggered condition-action rules for policy based management of networks and distributed systems. Ponder can also be used for security management activities such as registration of users or logging and auditing events for dealing with access to critical resources or security violations. Key concepts of the language include domains to group the object to which policies apply, roles to group policies relating to a position in an organisation [Lupu97], relationships to define interactions between roles and

management structures to define a configuration of roles and relationships pertaining to an organisational unit such as a department.

An obligation policy is depicted in the following figure:

```
inst oblig loginFailure {
on 3*loginfail(userid) ;
subject s = /NRegion/SecAdmin ;
target <userT> t = /NRegion/users ^ {userid} ;
do t.disable() -> s.log(userid) ;
}
```

This policy is triggered by 3 consecutive loginfail events with the same userid. The NRegion security administrator (SecAdmin) disables the user with userid in the /NRegion/users domain and then logs the failed userid by means of a local operation performed in the SecAdmin object. The -> operator is used to separate a sequence of actions in an obligation policy. Names are assigned to both the subject and the target. They can then be reused within the policy. In this example we use them to prefix the actions in order to indicate whether the action is on the interface of the target or local to the subject.

The policy description language (PDL) is an event-based language from Bell-Labs [Lobo99] in which they use the event-condition-action rule paradigm of active databases to define a policy as a function that maps a series of events into a set of actions. The language can be described as a real-time specialised production rule system to define policies. The syntax of PDL is simple and policies are described by a collection of two types of expressions: policy rules and policy defined event propositions. Policy rules are expressions of the form: *event causes action if condition* which reads: If the event occurs under the condition the action is executed. Policy defined event propositions are expressions of the form: *event triggers policy-defined-event if condition* which reads: If the event occurs under the condition, the policy-defined-event is triggered. Examples of policies expressed in PDL can be found in [Kohli99].

The path-based policy language (PPL) from the Naval postgraduate school described in [Stone01] is designed to support both the differentiated as well as the integrated services model and is based on the idea of providing better control over the traffic in a network by constraining the path (i.e. the links) the traffic must take. The rules of the language have the following format:

policyID <userID> @{paths} {target} {conditions} [{action_item}] action_item = [{condition}:] {actions}

Action_items in a PPL rule correspond to the if-condition-then-action rule of the IETF approach. The informal semantics of the rule is: policyID created by <userID> dictates that target class of traffic may use paths only if {conditions} is true after action_items are performed. The following are examples of PPL rules from [Stone01]:

Policy1 <net_manager> @ {<1,2,5>} {class = {faculty}} {*} {priority := 1}

Policy2 <Betty> @ {<1,*,5>} {traffic_class = {accounting}} {day != Friday : priority := 5}

Policy1 states that the path starting at node 1, traversing to node 2, and ending at node 5 will provide high priority for faculty users. Policy2 uses the wild-card character to specify a partial path. It states that, on all paths from node 1 to node 5, accounting class traffic will be lowered to priority 5 unless it is a Friday. In this policy the action_items field is used with temporal information to influence the priority of a class of traffic.

Other work on policy specification is carried out in the area of security and trust specification policy languages. These include: XACL which is an XML specification for expressing policies fro information access over the Internet and is being defined by the Organisation for the Advancement of Structured Information Standards (OASIS) technical committee, LaSCO [Hoag98] is a graphical approach for specifying security constraints on objects in which a policy consists of two parts: the domain and the requirement. The security policy language (SPL) [Ribe01] is an event driven policy language that supports access control, history-based and obligation-based policies.

10.10.4 Policy Information Models

As we mentioned earlier, IETF didn't specify a specific language to express network policies but rather a generic object-oriented information model for representing policy information following a rule-based approach i.e. if <condition> then <action>. This model is called the Policy Core Information Model (PCIM) [RFC3060] and extends the Common Information Model (CIM) defined by DMTF [DMTF] which defines generic objects such as managed system elements, logical and physical elements, systems, service, users, etc and provides abstractions and representations of the entities involved in a managed environments including their properties, operations and relationships.

The classes comprising the Policy Core Information Model are intended to serve as an extensible class hierarchy (through specialization) for defining policy objects that enable application developers, network administrators, and policy administrators to represent policies of different types. Each policy rule consists of a set of conditions and a set of actions. Policy rules may be aggregated into policy groups. These groups may be nested, to represent a hierarchy of policies. The set of conditions associated with a policy rule specifies when the policy rule is applicable. The set of conditions can be expressed as either an ORed set of ANDed sets of condition statements or an ANDed set of ORed sets of statements. Individual condition statements can also be negated. These combinations are termed, respectively, Disjunctive Normal Form (DNF) and Conjunctive Normal Form (CNF) for the conditions. If the set of conditions associated with a policy rule evaluates to TRUE, then a set of actions that either maintain the current state of the object or transition the object to a new state may be executed. For the set of actions associated with a policy rule, it is possible to specify an order of execution, as well as an indication of whether the order is required or merely recommended. It is also possible to indicate that the order in which the actions are executed does not matter. Policy rules themselves can be prioritised. One common reason for doing this is to express an overall policy that has a general case with a few specific exceptions. Moreover, policy conditions and policy actions can be partitioned into two groups: ones associated with a single policy rule, and ones that are reusable, in the sense that they may be associated with more than one policy rule. Conditions and actions in the first group are termed "rule-specific" conditions and actions; those in the second group are characterized as "reusable". Figure # shows the classes of PCIM and their main associations.

After PCIM became a standard track RFC, the IETF Policy Framework WG some changes to PCIM, which recently also became a RFC called the PCIM extensions (PCIMe) [RFC3460]. Two types of changes are included in PCIMe. First, several completely new elements are introduced, for example, classes for header filtering, that extend PCIM into areas that it did not previously cover. Second, there are cases where elements of PCIM (for example, policy rule priorities) are deprecated, and replacement elements are defined (in this case, priorities tied to associations that refer to policy rules). Both types of changes are done in such a way that, to the extent possible, interoperability with implementations of the original PCIM model is preserved. The PolicyRuleInPolicyRule and PolicyGroupInPolicyRule aggregations have been introduced in PCIMe. These aggregations make it possible to define larger "chunks" of reusable policy to place in a ReusablePolicyContainer. These aggregations also introduce new semantics representing the contextual implications of having one PolicyRule executing within the scope of another PolicyRule. Another major change from PCIM is the introduction of the Compound and Simple Policy Conditions and Actions. The idea is to create reusable "chunks" of policy that can exist as named elements in a ReusablePolicyContainer. The "Compound" classes and their associations incorporate the condition and action semantics that PCIM defined at the PolicyRule level: DNF/CNF for conditions, and ordering for actions. The SimplePolicyCondition / PolicyVariable / PolicyValue structure has been introduced into PCIMe. A list of PCIMe-level variables is defined, as well as a list of PCIMe-level values. Other variables and values may, if necessary, be defined in sub-models of PCIMe. For example, QPIM defines a set of implicit variables corresponding to fields in RSVP flows. A corresponding SimplePolicyAction / PolicyVariable / PolicyValue structure is also defined. While the semantics of a SimplePolicyCondition are "variable matches value", a SimplePolicyAction has the semantics "set variable to value".

The Policy Framework Group has also defined an information model for representing QoS network policies called QPIM [QPIM] based on the PCIM and PCIMe. QPIM build upon these two models to

define an information model for QoS enforcement for Differentiated and Integrated Services. For DiffServ QPIM provides actions and conditions that control the classification, policing and shaping done within the differentiated service domain boundaries, as well as actions that control the per-hop behaviour within the core of the DiffServ network, while for IntServ it provides actions that control the reservation of such requests within the network. Finally, IETF/DMTF defined a mapping of the Policy Core Information Model to a form that can be implemented in a directory that uses Lightweight Directory Access Protocol (LDAP) as its access protocol. This model defines two hierarchies of object classes: structural classes representing information for representing and controlling policy data as specified in RFC3060, and relationship classes that indicate how instances of the structural classes are related to each other. Classes are also added to the LDAP schema to improve the performance of a client's interactions with an LDAP server when the client is retrieving large amounts of policy-related information.

10.10.5 Policy execution/enforcement

While follows a more static approach for enforcing policies i.e. the Policy Decision Point translates the high-level policies to configuration commands supported by the devices e.g. SNMP, COPS-PR, CLI etc, the rest of the research community follows more dynamic programmable approach. In the Ponder deployment model the enforcement agents are called the policy management agents and responsible for enforcing the refrain and obligation policies. An overview of the operation of a policy management agent is shown in Figure 65:

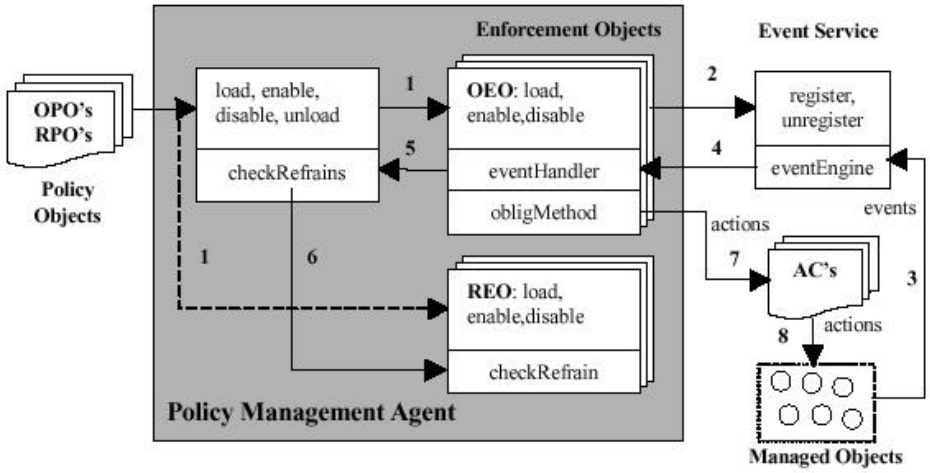


Figure 65 Overview of a Policy Management Agent

Policy management agents enforce all the enabled refrain and obligation methods for a subject. An overview of the operation of a policy management agent (PMA) is shown in figure 6. The enforcement objects for obligation policies and refrain policies (OEOs & REOs) are loaded from corresponding policy objects (OPOs & RPOs) and stored locally (1). When an obligation policy is enabled its obligation enforcement object registers the obligation event specification along with a reference to an event handler with the event service (2). The event service processes events (3) and disseminates them to handlers based on their event specifications (4). On receiving an event, handlers check both the constraints of the obligation policy and all enabled refrain enforcement objects (REOs) within the agent to check if any REO disallows actions within the obligation method (5 & 6). If constraints and refrains allow, the event handler then calls the obligation method, which performs actions on managed objects (7,8). Two interactions are omitted from figure 6. Firstly, the event handler in the PMA queries the domain service in order to evaluate the target set on which actions are to be invoked i.e., the event handler effectively coordinates the execution of the obligation policy. Secondly, obligation policies are allowed to invoke actions internal to the PMA.

[Martin02] uses the distributed management infrastructure as defined by the Script MIB can be used to control the distribution and execution of policy rules in a network with multiple PDPs, each realized

by an instance of the Script MIB. They present two approaches for realizing the policy execution engine. The first approach represents policies by program code. This matches the typical use of the Script MIB. A policy or a group of policies are represented by a program that is passed as a script to a Script MIB agent. At the agent the program is executed by a runtime engine for the used programming language. This runtime engine must provide a way of accessing the network elements to be configured by the policies, e.g. by offering a specific library. The second approach represents policies by objects. A policy or a group of policies are represented by a set of objects. Again, the set is passed as a script to a Script MIB agent. There, the objects are evaluated by a specific policy runtime engine. The objects representing policies conform to PCIM, they contain data only and no code. Also, they are specific to an application domain, e.g. QPIM [QPIM] for QoS. The policy runtime engine must contain implementations of all PCIM policy classes that are to be evaluated, and it must have access to the network elements to be configured by these policies.

A similar approach has been followed in [Flegk02] where policies are translated into scripts that are interpreted on the fly at a policy consumer point. The decomposition of the policy consumer is shown in Figure 66. The Policy Consumer component is decomposed in three parts. The first part, the *Repository Client* provides access to the Policy Storing Service, and is responsible for downloading the associated objects stored in PolSS that comprise the policy rules this specific policy consumer should enforce in order to influence the behaviour of the component it is attached to. The second part of the Policy Consumer is the *Script Generator*, which is responsible for creating the script that implements the policy. It contains logic, specific to the component that the policy consumer is attached to, that automates the process of generating a script from the higher-level representation of policies as they are stored in the PolSS. The *Policy Interpreter* provides the "glue" between the policy consumer and the policy-based component and interprets a language, which includes functions that perform management operations.

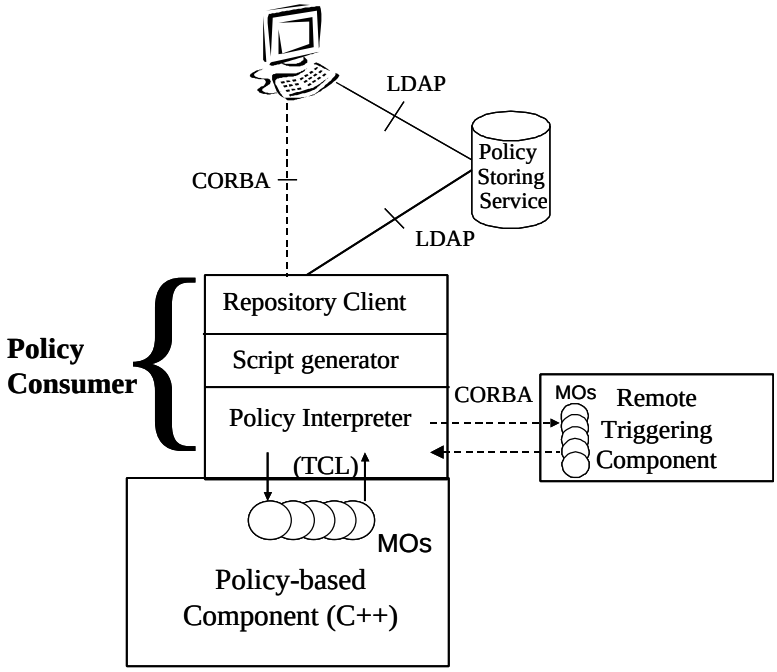


Figure 66 Policy Consumer Decomposition

10.10.6 The Common Open Policy Service Protocol (COPS)

The Common Open Policy Services (COPS) protocol and its extension for provisioning purposes (COPS-PR) are an interesting means to dynamically allow/release resources in a network to deploy a high-level service automation. This IETF proposition articulates itself around a client-server architecture, which potentially allows the configuration of any specific policy in the network, in so far

as the corresponding functional elements, which are to be exchanged between these two entities (aka. client and server), are defined.

10.10.6.1 COPS: standardisation aspects

10.10.6.1.1 Basis of the COPS framework

The COPS protocol specification has reached the status of "standard" in the IETF process since January 2000. The correspondent document, RFC 2748 [RFC-2748], defines the base protocol, the associated objects together with the subjacent model. This specification is resolutely and historically turned towards the use of COPS in an IntServ-RSVP context (*"outsourcing"* model). However, it is opened to other usages, as it is the case with COPS-PR. Inside the RAP (Resource Allocation Protocol) working group, the other documents describing the use of COPS for the outsourcing model and the required extensions are now standards.

Complementary, COPS for Provisioning (COPS-PR) is defined in the RFC 3084 [RFC-3084], which has reached the status of "standard" in the IETF process in March 2001. This document describes the necessary extensions for COPS to make it a provisioning tool (new objects have been specified in this aim). It also introduces the associated data model (the Policy Information Base commonly known as PIB), together with the general operational model.

Therefore, from the protocol definition point of view, the specifications are considered stable. The ongoing effort is concentrated on the definition of the PIBs, which are to be used with COPS-PR. The data structure used for the PIB definition, Structure of Policy for Provisioned Information (SPPI) is a standard defined in RFC 3159 [RFC-3159]. As far as the PIBs themselves are concerned, the main one, the framework PIB [FRWK-PIB] has reached the 9th version of draft and is in the IESG queue for a "last call". Other PIBs are under definition, either in the RAP WG, when it is relevant to the group charter, or in other IETF WG (but also external entities like 3GPP) when the topic tackled by the PIB is specific to a given domain (DiffServ, IPSec…).

10.10.6.1.2 QoS related specifications and propositions

The DiffServ WG within the IETF is in charge of the definition of the DiffServ PIB. This work is destined to describe the use of COPS-PR as a configuration protocol allowing the use of the necessary network resources in order to set up DiffServ-based QoS policies within a network. The current specification document [DIFF-PIB] has reached the 9th version of draft and has gone through a WG "last call". The status with which this document should be published in the coming months is not fixed at the moment. It is more likely to be edited as an informational or BCP (Best Current Practice) document.

Another proposition has been submitted regarding the use of COPS-PR for negotiating SLSes between entities. This individual submission is actually in version 3 of the draft, and has received a good feedback from the IETF community.

It is good to note that many individual submissions related to the use of COPS for QoS have been proposed, but unfortunately many have expired at the time of writing this document.

10.10.6.1.3 IP Traffic Engineering specifications and propositions

Many work items defining the use of COPS-PR in order to deploy IP traffic engineering policies have been submitted to the standardisation bodies.

A first document [IPTE-CT] defines a COPS Client-Type dedicated to IP traffic engineering. It presents the generic architecture, defines the content of the COPS messages (when used with this Client-Type) and the framework for this usage. This work is completed by another individual submission [IPTE-PIB], which specifies the PIB associated with the Client-Type defined within this context. At this stage, these two documents are still individual submissions and are proposed as experimental work. The corresponding drafts are respectively in version 3 and 2.

Moreover, another document comes to complement these specifications. It defines a new PIB [IPTE-ACC] which aim is to furnish accounting data. This information is destined to be used by the decision process relative to IP traffic engineering functions, and is gathered thanks to the reporting process of the IPTE Client-Type. This document is actually in version 1.

10.10.6.1.4 MPLS traffic engineering specifications and propositions

Regarding MPLS traffic engineering, the main work consists in defining a PIB, which has been specified in [PPVPN-PIB]. This PIB allows the dynamic allocation of resources associated with a virtual private network exploiting the resources of MPLS and BGP. In this context, COPS-PR enables the creation, configuration, monitoring and suppression of this VPN service. The actual version of the draft is 1.

Note that many individual submissions related to the use of COPS for MPLS traffic engineering have been proposed, but have expired at the time of writing this document.

10.10.6.2 Management Tools

Many commercial products exist in the market that provide policy-based network management solutions with most of them based on the IETF policy framework. Nortel's Optivity Policy Services is a system-level software application designed to manage the traffic prioritisation using DiffServ and network access security parameters for business applications in the enterprise-networking environment. The tool supports both Nortel BayRS and CISCO IOS router platforms. It uses LDAP to store policy information and it integrates both a COPS-PR stack and a COPS proxy-server, which translates the COPS commands to CLI syntax for the non-compliant COPS devices. The Juniper SDX-300 is a proprietary system that enables the configuration of Juniper's BAS ERX as far as DiffServ QoS policies are concerned. This tool should evolve in parallel with the functions embedded in the devices, i.e. the planned support of COPS standards instead of proprietary solutions. The Orchestream Enterprise solution is a Quality of Service management tool that leverages the DiffServ approach. Policies are specified using the IETF condition/action notation and can be stored in a LDAP. repository. In order to enhance the policy management capabilities, the Orchestream tool allows a network administrator to organise the devices to be managed in a hierarchy. This means that all the lower level devices will inherit policies specified below a given level. Like the Nortel solution, Orchestream supports COPS for sharing policy information with other nodes in the network. Additionally, devices can be configured using SNMP, HTTP and some other proprietary protocols. Policies can be triggered based on conditions that are specified using source/destination addresses, port numbers, IP protocol type, as well as on external events that are implemented through a custom software API. The tool does not allow conditions to be based on higher-level protocol information such as MAC addresses of VLAN tags. The more recent emphasis has been on network provisioning and integration with operational support systems rather than only policy-based management.

HP's PolicyXpert tool is a multi-platform policy based management solution, designed for integration into the company's Openview network management suite. In its current release, version 2.1, PolicyXpert supports traffic management actions ranging from priority marking to DiffServ code points. Like many of the other tools considered here, policies are defined using the if <condition> then <action> paradigm where conditions can be based on packet information, time of day or higher-level protocol information like HTTP URL or VLAN ID. The tool supports many of the prevalent standards, including COPS, DiffServ and RSVP. Cisco's policy based management offering, CiscoAssure Policy Manager, is also aimed at QoS service management. Although policies are specified using the condition/action approach defined by the IETF-CIM standard, the tool policies themselves are stored in a flat-file database [Conov99]. The user interface allows administrators to easily specify multiple conditions for triggering policies. Like the other tools considered here, conditions can be specified using a combination of IP addresses (source and destination), application ports, and the protocol being used (IP, TCP or UDP). Policy actions are applied to routers by using the Command Line Interface (CLI) language that is supported by Cisco hardware. Multi-vendor interoperability is provided with an implementation of COPS. In addition to supporting QoS related management operations, this tool allows the administrator to define access control policies for the

devices being managed. Finally, NetPolicy aims to provide policy based management capabilities for a range of Allot Communication's network hardware in addition to Cisco routers. However, results of tests performed on an early version of this tool concluded that the Cisco support was incomplete [Conover 1999]. Once again, policies are specified using the condition/action notation, and the conditions can be defined in terms of the packet information parameters mentioned previously. The policy repository is implemented using LDAP and policy information is passed to target devices using either COPS or CLI. Additionally, NetPolicy supports management operations on simple access control lists.

10.10.6.3 Hardware Devices

At the moment, no manufacturer has released a product delivering traffic engineering functionalities thanks to the use of COPS protocol. However, many devices already support the use of COPS for QoS configuration, using a DiffServ model.

The following devices propose COPS usage for QoS *a la DiffServ* configuration (non-exhaustive list):

- Cisco catalyst 5000, 6000 and 6500 series switches implement COPS for DiffServ configuration purposes based on a proprietary PIB. Concerning routers, no official IOS supports COPS functionalities for DiffServ, but some command line are available on the version 12.2(4)T1 for instance.
- Hitachi's GR2000 routers implement the COPS and COPS-PR in conformity with the standard RFCs. The implementation of COPS-PR is destined to the control of QoS policies *a la DiffServ* in these equipments.
- Nortel's range of products supporting the configuration of DiffServ QoS policies thanks to the use of COPS includes the Passport router family, the Baystack switches family and the Business Policy Switch switches family.
- Juniper proposes the support of COPS for QoS configuration on its ERX BAS product range. At the moment, this function relies on a proprietary solution, which should normally evolve towards the IETF standards during 2003.

10.11 References

[Abn01] Abarbanel, B., Venkatachalam, S., *BGP-4 support for Traffic Engineering*, draft-abarbanel-idr-bgp4-te-00.txt, September 2000

[Adams02] A. Adams et al, *Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification (Revised)*, draft-ietf-pim-dm-new-v2-02.txt, October 2002

[Ahuja93] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin, *Network Flows: Theory, Algorithms and Applications*, Prentice Hall, 1993

[ASP00] B. Jaruzelski, R. M. Lake, F. M. Riberio, ASP101: *Understanding the Application Service Provider Model*, Booz.Allen & Hamilton, 2000, www.bah.com.

[Aukia00] P. Aukia, et al., *RATES: A Server for MPLS Traffic Engineering*, IEEE Network Magazine, vol. 14, no. 2, pp. 34-41, March 2000

[Bain] A.Bain, P.Key, *Modelling the Performance of Distributed Admission Control for Adaptive Applications*

[Baker00] F. Baker, C. Iturralde, F. le Faucher and B. Davie, *Aggregation of RSVP for IPv4 and IPv6 Reservations*, Internet Draft, draft-ietf-issll-rsvp-aggr-02.txt, Mach 2000.

[Ballar97] A. Ballardie, *Core Based Trees (CBT version 2) Multicast Routing*, *RFC* 2189, September, 1997

[Bates98] T. Bates et al, *Multiprotocol Extensions to BGP-4*, *RFC* 2283, February 1998

[BEGD02] Bernet, Y., Elfassy, N., Gai, S. and D. Dutt, *RSVP Proxy*, draft-ietf-rsvp-proxy-03, expired Sept. 2002.

[BGRP] P. Pan, E, Hahne, and H. Schulzrinne, *BGRP: A Tree-Based Aggregation Protocol for Inter-domain Reservations*, Journal of Communications and Networks, Vol. 2, No. 2, June 2000, pp. 157-167.

[BGRP+] Stefano Salsano (ed.), *Inter-domain QoS Signalling: the BGRP Plus Architecture*, Internet Draft <draft-salsano-bgrpp-arch-00.txt>, Expired November, 2002.

[BGRP-fm] P. Pan, E. Hahne, H. Schulzrinne, *BGRP: A Framework for Scalable Resource Reservation*, Internet Draft <pan-bgrp-framework-00.txt> Bell Labs/Columbia Uni., Expired July, 2000.

[BGRP-per] Eugenia Nikolouzou, et al., BGRPP: *Performance evaluation of the proposed Quiet Grafting mechanisms*, Internet Draft <draft-nikolouzou-bgrpp-sim-00.txt>, Expired January 2003.

[Bhatt00] S. Bhattacharyya et al, *A Framework for Source-Specific IP Multicast Deployment*, draft-bhattach-pim-ssm-00.txt, July 2000

[Bianch03] G. Bianchi et al, *QUASIMODO: Quality of Service-aware Multicasting Over DiffServ and Overlay Networks*, IEEE Network, special issue on multicasting, January/February, 2003

[Blake98] S. Blake et al, *An Architecture for Differentiated Services*, *RFC* 2475, December 1998

[Bless02] R. Bless, K. Wehrle, *IP Multicast in Differentiated Services Networks*, <draft-bless-DiffServ-multicast-05.txt>, work in progress, November 2002

[Bon01] Bonaventure, O., *Using BGP to distribute flexible QoS information*, draft-bonaventure-bgp-qos-00.txt, February 2001

[Breit02] Y. Breitbart, M. Garofalakis, A. Kumar and R. Rastogi, *Optimal Configuration of OSPF Aggregates*, In Proc. of IEEE INFOCOM02, New York, USA, June 2002

[Bresl00] L. Breslau, et al., *Endpoint Admission Control: Architectural Issues and Performance*, ACM Sigcomm 2000

[Camarillo02] Camarillo G. et al., *Integration of Resource Mangement and SIP*, IETF Internet Draft <draft-ietf-sip-manyfolds-resource-07.text>, April 2002

[Cao00] Z. Cao, Z. Wang, and E. Zegura, *Performance of Hashing-based Schemes for Internet Load Balancing*, In Proc. of IEEE INFOCOM 00, pp. 332-341, March 2000

[Carlb97] K. Carlberg et al, *Building Shared Trees Using a One-to-many joining Mechanism, ACM* Computer Communication Review, January 1997, pp5-11

[CDN01] B. Krishnamurthy, C. Wills, Y. Zhang, *On the Use and Performance of Content Distribution Networks*, ACM SIGCOMM Internet Measurement Workshop 2001.

[Centin00] C. Centinkaya and E. Knightly, *Scalable Services via Egress Admission Control*, In Proceedings of IEEE INFOCOM'00, Tel Aviv, Israel, March 2000.

[Centink] C.Cetinkaya, E.W. Knightly, *Egress Admission Control*

[Chen00] S. Chen et al, *A QoS-Aware Multicast Routing Protocol*, in proc. IEEE INFOCOM 2000, Vol. 3 pp.1594-1603

[Chen98] S. Chen, K. Nahrstedt, *An Overview of Quality-of-Service Routing for the Next Generation High-Speed Networks: Problems and Solution*", IEEE Network Magazine, vol. 12, no. 6, pp. 64-79, November 1998

[Choi01] B. Kyu Choi, R.Bettati, *Endpoint Admission Control: Network Based Approach*, 21 International Conf. On Distributed Computing Systems, Phoenix, April 2001

[Cisco] Cisco, Cisco IOS Software Documentation, *New Features in Release 12.0(22)S*

[Conov99] Conover, J., *Policy-based Network Management*, Network Computing, http://www.networkcomputing.com/1024/1024f1.html, 1999

[Cri01] G. Cristallo, C. Jacquenet, *Providing Quality of Service Indication by the BGP-4 Protocol: the QOS_NLRI attribute*, <draft-jacquenet-qos-nlri-04.txt>, March 2002

[Damian01] Damianou, N., Dulay, N., et al., *The Ponder Policy Specification Langauge*, in Proc. of the IEEE Workshop on Policies for Distributed Systems and Networks (Policy 2001), Bristol, U.K., January 2001

[Dee01] S. E. Deering, *Multicast Routing in a Datagram Internetwork*, Ph.D. thesis, Stanford University, Dec 1991

[Deering89] S. Deering et al, *Host Extensions for IP Multicasting*, *RFC* 1112, Aug. 1989

[Deering96] S. Deering et al, *The PIM Architecture for Wide-Area Multicast Routing*, *IEEE/ACM* Transactions on Networking, Vol. 4, No. 2, Apr. 1996, pp 153-162

[Devalla] B.Devalla et al., *Adaptive Connection Admission Control for Mission Critical Real-Time Communication Networks*

[DIFF-PIB] M. Fine, K. McCloghrie, J. Seligson, K. Chan, S. Hahn, C. Bell, A. Smith, F. Reichmeyer, *Differentiated Services Quality of Service Policy Information Base*, draft-ietf-DiffServ-pib-09.txt, June 2002

[DMTF] http://www.dmtf.org

[DRAFT-FLOWLABEL] J. Rajahalme, A. Conta, B. Carpenter, S. Deering, *IPv6 Flow Label Specification*, draft-ietf-ipv6-flow-label-04.txt, December 2002

[DRAFT-MBGP] T. Bates, R. Chandra, D. Katz, Y. Rekhter, *Multiprotocol Extensions for BGP-*, draft-ietf-idr-rfc2858bis-02.txt

[DRAFT-QOS-FLOW] H. Jagadeesan, T. Singh, *A Radical Approach in providing Quality-of-Service over the Internet using the 20-bit IPv6 Flow Label field*, draft-jagadeesan-rad-approach-service-01.txt, March 2002

[Elek00] V. Elek et al., *Admission Control Based on End-to-End Measurements*, In Proc. of IEEE INFOCOM'00, Tel Aviv, Israel, March 2000.

[Elwal01] A. Elwalid, C. Jin, S H. Low, and I. Widjaja, *MATE: MPLS Adaptive Traffic Engineering*, In Proc. of IEEE INFOCOM2001, pp. 1300-1309, Alaska, USA, April 2001

[ETSI] European Telecommu), ETSI ES 201 915-1 *Open Service Access; Application Programming Interface (API); Part 1: Overview*, related standards available at http://portal.etsi.org

[Falou98] M. Faloutsos *et al*, *QoSMIC*: *Quality of Service Sensitive Multicast Internet protocol*, proc. *ACM SIGCOMM* 1998, pp144-153

[Farina98] D. Farinacci et al, *Multicast Source Discovery Protocol (MSDP)*, Internet Draft, draft-farinacci-msdp-*.txt, Jul. 1998

[Fauch02] F. Le Faucheur, et al. *Requirements for support of Diff-Serv-aware MPLS Traffic Engineering*, IETF Internet draft, draft-ietf-tewg-diff-te-reqts-05.txt, work in progress, June 2002

[Feld01] A. Feldmann and J. Rexford, *IP Network Configuration for Intradomain Traffic Engineering*, IEEE Network Magazine, vol. 15, no. 5, pp. 46-57, September 2001

[Fenner02] B. Fenner, *Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)*, draft-ietf-pim-sm-v2-new-06.txt, December 2002

[Fenner97] W. Fenner, *Internet Group management Protocol, version 2*, *RFC* 2236, Nov. 1997

[Flegk02] Flegkas, P., Trimintzios, P., Pavlou, G., *A Policy-based QoS Management System for IP DiffServ Networks*, IEEE Network Magazine, March/April 2002

[Fotz00] B. Fortz, and M. Thorup, *Internet Traffic Engineering by Optimizing {OSPF} Weights*, In Proc. of IEEE INFOCOM 2000, pp. 519-528, Israel, March 2000

[FRWK-PIB] M. Fine, K. McCloghrie, J. Seligson, K. Chan, R. Sahita, S. Hahn, A. Smith, F. Reichmeyer, *Framework Policy Information Base*, draft-ietf-rap-frameworkpib-09.txt, June 2002

[Fu02] Xiaoming Fu, et al, *Analysis on RSVP Regarding Multicast*, Internet Draft, draft-fu-rsvp-multicast-analysis-01.txt), Expires April 2003.

[Gibbens99] R.J. Gibbens, F.P. Kelly, *Distributed connection acceptance control for a connectionless network*, 16th International Teletraffic Congress, Edinburgh, June 1999

[Hanna99] S. Hanna, B. Patel, M. Shah, *Multicast Address Dynamic Client Allocation Protocol (MADCAP)*, *RFC* 2730, Dec. 1999

[Hoag98] Hoagland, J. A., Padney, R., et al., *Security Policy Specification using a Graphical Approach*, UC Davis, Computer Science Department

[Holbro02] H. W. Holbrook, *Using IGMPv3 and MLDv2 For Source-Specific Multicast*, draft-holbrook-idmr-igmpv3-ssm-03.txt, November 2002

[Holbro99] H. W. Holbrook, D. R. Cheriton, *IP Multicast Channels: EXPRESS Support for Large-scale Single-source Applications*, Proc. *ACM SIGCOMM'99*

[IETF] Internet Engineering Task Force (IETF www.ietf.org, related work groups and drafts).

[IPDR] ipdr.org, *Service Specifications*, related documents, http://www.ipdr.org

[IPTE-ACC]M. Boucadair, *An IP Traffic Engineering PIB for Accounting purposes"*, "draft-boucadair-ipte-acct-pib-01.txt, December 2002

[IPTE-CT] C. Jacquenet, *A COPS client-type for IP traffic engineering*, draft-jacquenet-ip-te-cops-04.txt, January 2003

[IPTE-PIB] M. Boucadair, C. Jacquenet, *An IP Traffic Engineering Policy Information Base*, draft-jacquenet-ip-te-pib-02.txt, June 2002

[Ish01] K. Ishiguro, T. Takada, *Traffic Engineering Extensions to OSPFv3"*, "draft-ishiguro-ospfv3-01.txt, October 2002

[Ivars] I.M. Ivars, G. Karlsson, *PBAC: Probe-Based Admission Control*

[Jain] The Parlay Group, *The Jain APIs: Integrated May 2002*, related documents available at http://java.sun.com/products/jain/

[Jia97] X. Jia et al, *Group Multicast Routing Algorithm by Using Multiple Minimum Steiner Trees*, Computer Communications 20 (1997) pp750-758

[Jia98] X. Jia, *A Distributed Algorithm of Delay-bounded Multicast Routing for Multimedia Applications in Wide Area Networks*, IEEE/ACM Transactions on Networking 6(6): 828-837 (1998)

[Johnston03] Johnston A. et al, *Session Initiation Protocol Private Extension for an OSP Authorization Token*, IETF Internet Draft draft-johnston-sip-osp-token-04.txt, February 2003

[Juniper] Juniper, *Junos software documentation*

[Karlsson] G. Karlsson, F. Orava, *The DIY Approach to QoS*

[Kat02] D. Katz, D., Yeung, K., Kompella, *Traffic Engineering Extensions to OSPFv2*, draft-katz-yeung-ospf-traffic-09.txt, October 2002

[Kau02] Kaur, H., Kalyanaraman, S., *A Connectionless Approach to Intra- and Inter-Domain Traffic Engineering*, 2nd New York Metro Area Networking Workshop, September 2002.

[Kelly] F.P. Kelly, P.B. Key, S. Zachary, *Distributed Admission Control*

[Knight99] E.W. Kinghtly, N.B. Shoft, *Admission Control for Statistical QoS: Theory and Practice*, IEEE Network, March 1999

[Kodi00] M. Kodialam, and T.V. Lakshman, *Minimum Interference Routing with Applications to Traffic Engineering*, in Proc. IEEE INFOCOM00, pp. 884-893 March 2000

[Kohli99] Kohli, M., Lobo, J., *Policy-based Management of telecommunication Networks*, in the Proc. of IEEE Policy Workshop (Policy99), HP Labs, Bristol, U.K., 1999

[Kompe93] V. P. Kompella et al, *Two Distributed Algorithms for multicasting multimedia Information*, Proc. *IEEE ICCCN'1993*, pp343-349

[Kou81] L. Kou et al, *A Fast Algorithm for Steiner Trees*, Acta Inormatica 15 (1981) pp141-145

[KPP93] V. P. Kompella et al, *Multicast Routing for Multimedia Communication*, IEEE/ACM Transaction on Network 1993, pp286-292

[Kummar99]S. Kummar et al, *The MASC/BGMP Architecture for Inter-domain Multicast Routing*, Proc. *ACM SIGCOMM'99*

[Lobo99] Lobo, J., Bhatia, R., et al., *A Policy Description Language*, in Proc. of AAAI, Orlando, Florida, 1999

[Low00] C. P. Low et al, *An Efficient Algorithm for Group Multicast Routing Problem with Bandwidth Reservations*, Computer Communications, Vol. 23(18) (2000) pp1740-1746.

[Low02] C. P. Low et al, *On Finding Feasible Solutions for the Delay Constrained Group Multicast Routing Problem*, *IEEE* Transactions on computers, vol. 51 No. 5 (2002) pp581-588

[Lupu97] Lupu, E., Sloman, M., *Towards a Role Based Framework for Distributed Systems Management*, Journal of Networks and Systems Management (JNSM), Vol. 5, no. 1, 1997

[Marr96] Marriot, D., Sloman, M., *Implementation of a Management Agent for Interpreting Obligation Policy*, in Proc. of the seventh IFIP/IEEE International Workshop on Distributed Systems: Operations & Management (DSOM'96), L'Aquila, Italy, October 28-30, 1996.

[Martin02] Martinez, P., Brunner, M., et al., *Using the Script Mib for Policy-based Configuration Management*, in the Proc of the IEEE Network Operations and Management Symposium (NOMS'02), Florence, Italy, April 2002.

[Mayer00] D. Mayer, P. Lothberg, *GLOP Addressing in 233/8*, *RFC* 2770, Feb. 2000

[Mitra99a] D. Mitra, and K. G. Ramakrishnan, *A Case Study of Multiservice, Multipriority Traffic Engineering Design for Data Networks*, In Proc. IEEE GLOBECOM 99, pp. 1087-1093, Brazil, December 1999

[Mitra99b] D. Mitra, J.A. Morrison and K.G. Ramakrishnan, *Virtual Private Networks: Joint Resource Allocation and Routing Design*, In Proc. IEEE INFOCOM 99, USA, March 1999

[Mortier00] R. Mortier, et al., *Implicit Admission Control*, IEEE Journal on selected areas in communications, vol. 18, no. 12, December 2000

[Moy94] J. Moy, *Multicast Extensions to OSPF*, *RFC* 1584, Mar. 1994

[MSK+02] J. Manner, T. Suihko, M. Kojo, M. Liljeberg, K. Raatikainen, *Localized RSVP*. Internet Draft draft-manner-lrsvp-01.txt, Expires July 2003.

[Mykon03] E.Mykoniati et al., *Admission Control for Providing QoS in DiffServ IP Networks: The TEQUILA approach* IEEE Communications Magazine, January 2003

[Pelsser02] Cristel Pelsser (FUNDP), Olivier Bonaventure (UCL), *RSVP-TE extensions for inter-domain LSPs*, Internet draft, Work in Progress, October 2002.

[PolicyWG] http://www.ietf.org/html.charters/policy-charter.html

[Poppe00] F. Poppe, et al. *Choosing the Objectives for Traffic Engineering in IP Backbone Networks Based on Quality-of-Service Requirements*, In Proc. Workshop on Quality of future Internet Services (QofIS'00), pp. 129-140, Germany, September 2000

[PPVPN-PIB] Y. El Mghazli, *BGP/MPLS VPN Policy Information Base*, draft-yacine-ppvpn-2547bis-pib-01.txt, July 2002

[QPIM] Snir, Y., Ramberg, Y., et al., *Policy QoS Information Model*, internet-draft, IETF, Novemeber 2001

[RAPWG] http://www.ietf.org/html.charters/rap-charter.html

[RFC 2597] J. Heinanen, et. el., *Assured Forwarding PHB Group*, June 1999.

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

[RFC 3086] K. Nichols, B. Carpenter, *Definition of Differentiated Services Per Domain Behaviors and Rules for their Specification*, April 2001.

[RFC 3140] S. Brim, B. Carpenter, F. Le Faucheur, *Per Hop Behavior Identification Codes*, June 2001.

[RFC 3246] B. Davie, et al, *An Expedited Forwarding PHB (Per-Hop Behavior)*, March 2002.

[RFC 3317] K. Chan, R. Sahita, S. Hahn, K. McCloghrie, *Differentiated Services Quality of Service Policy*, Informational RFC, March 2003.

[RFC1771] Y. Rekhter, T. Li, *A border gateway protocol 4 (BGP-4)*, RFC 1771, March 1995

[RFC2205] Braden, R., Zhang, L., Berson, S., Herzog, S. and S. Jamin, *Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification*, RFC 2205, Sep 1997.

[RFC2207] L. Berger and T. O'Malley, *RSVP Extensions for IPSEC Data Flows*, RFC 2207, September 1997.

[RFC2328] J., Moy, *OSPF Version 2*, RFC 2328, April 1998.

[RFC2380] Berger, L., *RSVP over ATM Implementation Requirements*, RFC 2380, August 1998.

[RFC-2460] S. Deering, R. Hinden, *Internet Protocol, Version 6 (IPv6) Specification*, RFC 2460, Standards Track, December 1998

[RFC-2474] K. Nichols, et. al., *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*, December 1998

[RFC2475] S. Blake, et. al., *An Architecture for Differentiated Services*, December 1998.

[RFC2573] Yavatkar, R., Pendarakis, D., Guerin, D., *A Framework for Policy Based Admission Control*, Informational RFC 2753, January 2000

[RFC2702] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, J. McManus, *Requirements for Traffic Engineering Over MPLS*, RFC 2702, September 1999.

[RFC2740] R. Coltun, D. Ferguson, J.Moy, *OSPF for IPv6*, RFC2740, December 1999.

[RFC2745] Terzis, A., Braden B., S. Vincent, and L. Zhang, *RSVP Diagnostic Messages*, RFC 2745, January 2000.

[RFC2746] Terzis, A., Krawczyk, J., Wroclawski, J. and L. Zhang, *RSVP Operation Over IP Tunnels*, RFC 2746, January 2000.

[RFC2748] Durham D., et al, *The COPS (Common Open Policy Service) Protocol*, IETF RFC 2748, January 2000

[RFC2814] Yavatkar, R., Hoffman, D., Bernet, Y., Baker, F. and M. Speer, *SBM (Subnet Bandwidth Manager): A Protocol for Admission Control over IEEE 802-style Networks*, RFC 2814, May 2000.

[RFC2842] R. Chandra, J. Scudder, *Capabilities Advertisement with BGP-4*, RFC 2842, Standards Track, May 2000

[RFC2842] R. Chandra, J. Scudder, *Capabilities Advertisement with BGP-4*, RFC 2842, Standards Track, May 2000

[RFC2961] Berger, L., Gan, D., Swallow, G., Pan, P. and F. Tommasi, *RSVP Refresh Reduction Extensions*, RFC 2961, April 2001.

[RFC2996] Bernet, Y., *Format of the RSVP DCLASS Object*, RFC 2996, November 2000.

[RFC3060] Moore, B., Elleson, E., et al., *Policy Core Information Model – Version 1 Specification*, Standard-Tracks RFC 3060, IETF, February 2001.

[RFC-3084] K. Chan, J. Seligson, D. Durham, S. Gai, K. McCloghrie, S. Herzog, F. Reichmeyer, R. Yavatkar and A. Smith, *COPS Usage for Policy Provisioning*, RFC 3084, March 2001

[RFC3107] Y. Rekhter (Juniper Networks), E. Rosen (Cisco Systems, Inc.), *Carrying Label Information in BGP-4*, Network Working Group, RFC 3107, IETF, May 2001.

[RFC-3159] K. McCloghrie, M. Fine, J. Seligson, K. Chan, S. Hahn, R. Sahita, A. Smith, F. Reichmeyer, *Structure of Policy Provisioning Information (SPPI)*, RFC 3159, August 2001

[RFC3175] F. Baker, C. Iturralde, F. Le Faucheur, B. Davie, *Aggregation of RSVP for IPv4 and IPv6 Reservations*, RFC 3175, Sept. 2001.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V. and G. Swallow, *Extensions to RSVP for LSP Tunnels*, RFC 3209, Dec. 2001.

[RFC3219] Rosenberg J. et al, *Telephone Routing over IP (TRIP),* IETF RFC 3219, January 2002

[RFC3261] Rosenberg J. et al, *SIP: Session Initiation Protocol,* IETF RFC 3261, June 2002

[RFC3270] F. Le Faucheur (ed), L. Wu, and et al, *Multi-Protocol Label Switching (MPLS) Support of Differentiated Services*, RFC 3270, May 2002.

[RFC3272] D. Awduche, et al. *Overview and Principles of Internet Traffic Engineering*, IETF Informational RFC-3272, May 2002

[RFC3460] Moore, B., *Policy Core Information Model Extensions (PCIMe)*, Standards-Track RFC 3460, IETF, January 2003

[Ribe01] Ribeiro, C., Zuquete, A., et al., *SPL: An access control language for security policies with complex constraints*, in the Proc. of Network and Distributed System Security Symposium (NDSS'01), San Diego, California, 2001

[Rie01] A., Riedl, D., Schupke, *A Flow-Based Approach for IP Traffic Engineering Utilizing Routing Protocols With Multiple Metric Types*, Sixth INFORMS Telecommunications Conference, March 2002

[Rouska97] G. N. Rouskas, et al, *Multicast Routing with End-to-end Delay and Delay Variation Constraints*, *IEEE* Journal on Selected Areas in Communications 15(3): 346-356 (1997)

[RSVP] Zhang, L., Deering, S., Estrin, D. and D. Zappala, *RSVP: A New Resource Reservation Protocol*, IEEE Network, Volume 7, Pages 8-18, Sep 1993.

[Salsano01] Salsano S., *COPS Usage for DiffServ Resource Allocation (COPS-DRA*", IETF Internet Draft <draft-salsano-cops-dra-00.txt>, October 2001

[Sargen] Susana Sargento et al., *Call Admission Control in IP networks with QoS support*

[SIBBS] QBone Signalling Design Team - *Final report*, http://qos.internet2.edu/wg/documents-informational/20020709-chimento-etal-qbone-signaling/

[SIG1] Manner (ed.), J. X. Fu, *Analysis of Existing Quality of Service Signalling Protocols*, <draft-ietf-nsis-signalling-analysis-00.txt> Expires April, 2003.

[SIG2] H.de meer, et al., *Analysis of Existing QoS solutions*, <draft-demeer-nsis-analysis-03.txt>, Expires May 2003.

[Sinnreich00] Sinnreich H., Donovan S., Rawlins D., *Inter-domain IP Communications with Qos, Authroization and Usage Reporting*, IETF Internet Draft <draft-sinnreich-sip-qos-osp-01.txt>

[Slom94] Sloman, M., *Policy Driven Management For Distributed Systems*, Journal of Network and Systems Management, Vol. 2, No. 4, pp. 333-360, Plenum Publishing, December 1994

[Slom99] Sloman, M., Lupu, E., *Policy Specification for Programmable Networks*, in Proc. of the 1st International Conference on Active Networks, Berlin, Germany, June 1999

[Stoica99] I. Stoica and H. Zhang, *Providing Guaranteed Services without per Flow Management*, in Proceedings of ACM SIGCOMM'99, Cambridge, MA, August 1999.

[Stone01] Stone, G. N., Lundy, B., et al., *Network Policy Languages: A Survey and a New Approach*, IEEE Network Magazine, pp 10-21, Jan/Feb 2001

[Stream01] D. Wu, et. al., *Streaming Video, over the Internet: approaches and Directions*, IEEE Transcations on circuits and systems for video technology, Vol. 11, No. 1, Feb 2001.

[Strieg01] A. Striegel, G. Manimaran, *A scalable approach for DiffServ Multicasting*, proc. *IEEE ICC* 2001

[Suri01] S. Suri, et al., *Profile-based Routing: A New Framework for MPLS Traffic Engineering*, In Proc. of the 2nd International Workshop on Quality of future Internet Services (QofIS'01), pp. 138-157, Portugal, September 2001

[Taka80] H. Takahashi, A Mastuyama, *An Approximate Solution for the Steiner Problem in Graphs*, Math. Japonica 6, pp573-577

[Tequila] IST project TEQUILA, http://www.ist-tequila.org

[Thaler00] D. Thaler et al, *The Internet Multicast Address Allocation Architecture, RFC* 2908, Sept. 2000

[Thaler02] D. Thaler, *Border Gateway Multicast Protocol (BGMP): Protocol Specification*, draft-ietf-bgmp-spec-03.txt, July 2002

[Thom01] Michael Thomas, *Analysis of Mobile IP and RSVP Interactions*, draft-thomas-seamoby-rsvp-analysis-00.txt, Issued Oct. 2002.

[TINA] Telecommunications Information Network Architecture (TINA) Consortium, related documents available at www.tinac.org, 1996-2000.

[TMF] TeleManagement Forum, *Telecom Operations Map*, March 2000; HYPERLINK "http://www.tmforum.org" www.tmforum.org

[TMN] ITU-T TMN Recommendation M.3400 *TMN Telecommunication Management Network*, M.3200 TMN Management Services, and related recommendations.

[Trimin01] P. Trimintzios et al., *A Management and Control Architecture for Providing IP Differentiated Services in MPLS-based Networks*, IEEE Commun. Mag., vol. 39, no. 5, May 2001.

[Tsch02] Hannes Tschofenig, *RSVP Security Properties*, Internet Draft draft-tschofenig-rsvp-sec-properties-00.txt, Expired Nov. 2002.

[Uhlig00] Steve Uhlig, Olivier Bonaventure, *On the Cost of using MPLS for inter-domain traffic*, COST263 workshop, September 2000.

[Vasseur00] J.P.Vasseur, et. al., *Definition of an RRO node-id subobject*, Internet Draft, draft-vasseur-mpls-nodeid-subobject-00.txt, Work in Progress, February 2003

[Vasseur01] J.P. Vasseur, Y. Ikejiri, *Reoptimization of an explicit loosely routed MPLS TE paths*, Internet Draft, draft-vasseur-mpls-loose-path-reopt-01.txt, Work in Progress, February 2003

[Vasseur02] J.P. Vasseur, et. al., *RSVP Path computation request and reply messages*, Internet Draft, draft-vasseur-mpls-computation-rsvp-03.txt, June 2002

[Vasseur03] Jean-Philippe Vasseur, Raymond Zhang, *Inter-AS MPLS Traffic Engineering*, Internet Draft, draft-vasseur-inter-as-te-00.txt, Work in Progress, February, 2003

[Veltri02] Veltri, L., Salsano., S and Papalilo, D., *SIP Extensions for QoS Support*, IETF Internet draft <draft-veltri-sip-qsip-01.txt> , October 2002

[Vida02] R.Vida et al, *Multicast Listener Discovery Version 2 (MLDv2) for IPv6*, draft-vida-mld-v2-06.txt, November 2002

[W3C] World Wide Web Scenarios, June 2002; related documents available at http://www.w3.org

[Waitz88] D. Waitzman, C. Partridge, S. Deering, *Distance Vector Multicast Routing Protocol (DVMRP)*, *RFC* 1075, Nov. 1988

[Wang01] Z. Wang, Y. Wang, and L. Zhang, *Internet Traffic Engineering without Full Mesh Overlaying*, In Proc. of IEEE INFOCOM 2001, Alaska, April 2001

[Xia01] Xiao, L., Shan-Lui, K., Wang, J., Nahrstedt, K., *QoS Extension to BGP*, UIUCDCS-R-2002-2295, September 2002.

[Xuan] D. Xuan et al., *Utilization-Based Admission Control for Real-Time Applications*

[Yang01] D. Yang et al, *MQ: An Integrated Mechanism for Multimedia Multicasting*, IEEE Transactions on multimedia, vol.3, no.1, March 2001

[Zhang] Zhi-Li Zhang et al., *Decoupling QoS Control from Core Routers: A novel Bandwidth Broker architecture for scalable support of Guaranteed Services*

[Zhang03] Raymond Zhang, JP Vasseur, *MPLS Inter-AS Traffic Engineering requirements*, Internet draft, draft-zhang-mpls-interas-te-req-02.txt, Work in Progress, February 2003

[Zhu95] Q. Zhu, et al, *A Source Based Algorithm for Delay-constrained Minimum-cost Multicasting*, in proc. of IEEE INFOCOM'95