QoS architecture for Broadband Satellite Multimedia (BSM) networks

R. J. Mort[1], S. Combes[2] , H. Cruickshank[3]

***Abstract*—. The objective of this paper is to identify a BSM QoS functional architecture applicable to a range of QoS scenarios, and capable of integration within next generation networks (NGN's). A modular, building block approach to network QoS provision is described which can be adapted to the performance and resulting complexity needed**

QoS mechanisms such as DiffServ, IntServ and NSIS and also service layer protocols are included.

The ETSI BSM architecture is characterised by the separation between common Satellite-Independent (SI) protocol layers and alternative lower Satellite-Dependent (SD) layers. A centralised QoS/resource management approach for the satellite network proposed, at both SI and SD layers.

***Index Terms*—QoS, DiffServ, IntServ, ETSI, BSM.**

I. INTRODUCTION

IP-based services and their users are growing ever more sophisticated, and QoS is a feature which will be increasingly valuable for service differentiation and support of more QoS-sensitive applications. In contrast to wired or optical networks where over-provisioning of capacity is often used to ensure QoS for packet-based transport, satellite systems, as for other wireless networks, allocate capacity efficiently and carefully. This requires more sophisticated QoS methods that are closely linked to resource provision and control at lower protocol layers than IP.

QoS provision within ETSI BSM systems [1] is one of the first aims, but since BSM systems are intended to access the Internet, end-to-end QoS across integrated networks including satellites is also important.

A BSM QoS functional architecture has been defined for IP-based applications. Compatibility with QoS requirements for generic internetworking including Next Generation Networks (NGN's, [4], [5]) are taken into account.

The BSM architecture is characterised by the separation between common Satellite-Independent (SI) protocol layers and alternative lower Satellite-Dependent (SD) layers [2]. At the SI layers, several methods of ensuring end-to-end QoS over integrated networks are foreseen, by means of signalling protocols (e.g. based on SIP) at the session (or application) layers and DiffServ, RSVP/IntServ, or NSIS at the IP layer. At the SD Layers alternative lower protocol layers offer different QoS characteristics. The focus of the architecture definition here is on maintaining compatibility with these alternative methods and approaches by addressing the generic BSM QoS functions required in the SI layers (including IP). These functions will provide interfaces where appropriate with higher-layer and lower-layer QoS functions, and with external networks and customer equipment.

II. QOS ARCHITECTURE APPROACH FOR IP NETWORKS

Overall approaches to QoS architectures in emerging and future IP-based networks have been described in IETF [3], ETSI TISPAN (NGN) [5], ITU-T SG13 [6], ADSL Forum[7], 3GPP[7], etc..

One of the common characteristics of these approaches is the uncoupling of services and networks, allowing services and networks to be offered separately and to evolve independently. Therefore in the architectures described there is a clear separation between the functions for services and for transport, and an open interface should be provided between them. Provisioning of existing and new services can then be independent of the network and the access technology.

In emerging networks (such as NGN's) there is increased emphasis by Service Providers on service customisation by their customers by means of service related APIs (Application Programming Interfaces) in order to support the creation, provisioning and management of services. In such networks the functional entities controlling policy, sessions, media, resources, service delivery, security, etc. may be distributed over the infrastructure, including both existing and new networks. When they are physically distributed they should communicate over open interfaces.

The approach is to differentiate between the Application (Service) and Transport Strata in Figure 1.

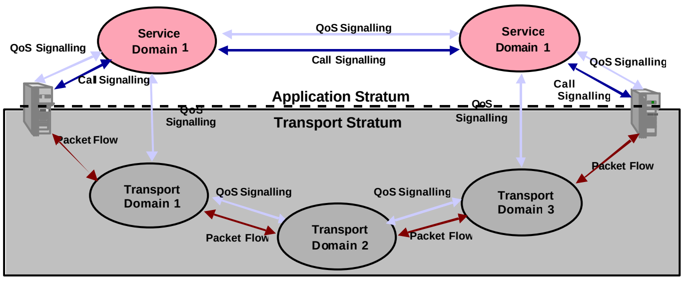

Figure 1– NGN Application & Transport Strata

The Application Stratum provides the service to users. Service is requested by user/call signalling protocols (e.g. H225+H245, SIP+SDP, H248).

The Transport Stratum provides a packet-oriented transport service and the desired network QoS. The QoS is requested by QoS signalling protocols (e.g. RSVP, COPS, NSIS). QoS signalling can be exchanged with user endpoints and/or the application plane.

Although QoS signalling is not always required, for example where provisioning combined with Diffserv networks are employed, it will generally be needed for guaranteed services.

Two general models of service assurance can be applied to user and network QoS: guaranteed and relative QoS (corresponding approximately to the Intserv and Diffserv models).

1 Systek Consulting Ltd., Havant, UK
2 ESTEC, Noordwijk, NL
3 CCSR, University of Surrey, Guildford, UK

The ways in which these models are implemented across the network including the BSM subnetwork are fundamentally different and will have an impact on the QoS architecture. However, IntServ and DiffServ can co-exist as DiffServ Domains should pass IntServ reservation requests transparently. Both IntServ and DiffServ models can also be driven from a policy database.

A. QoS Network Building Blocks

To offer QoS services outlined above in a complete and efficient way can be complex, and can involve multiple inter-related aspects. For example, in case of network resource contention or congestion, to maintain the expected service response requires a variety of functions working at different time-scales, ranging from careful network planning based on traffic patterns over a long period (grouped in the Management Plane) to differential resource allocation and admission control based on the current network load conditions (in the Control Plane).

The range of mechanisms involved in QoS can be considered as a set of building blocks, or functions which can be combined in different ways to provide different overall objectives (e.g. type of network, or for guaranteed or relative QoS). These building blocks, to be provided by one or more network service providers for the benefit of users, may be classified in the Management, Control and Data Planes as shown in Figure 2.

A comprehensive QoS solution typically employs multiple building blocks across the Management Plane, Control Plane and Data Plane, but practical implementations may require only a subset of the functions.

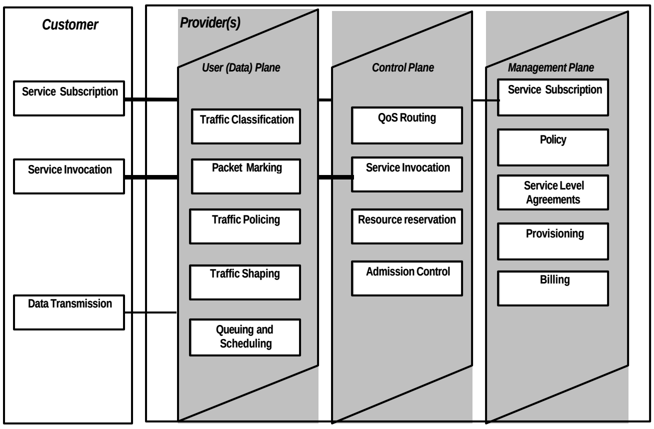

Figure 2– QoS Building Blocks

B. Interactions between building blocks

QoS parameters need to be exchanged between the various building blocks. These parameters include transaction performance at the packet level (e.g., delay and packet loss) and service reliability/availability expectations in the form of traffic priority levels for specific network functions such as admission control and traffic restoration. Examples for mechanisms to convey these parameter values are signalling and database lookups and include:

- QoS signalling: signalling of QoS parameter requirements per service/flow between functional blocks
- Call Signalling: service invocation, resource reservation
- Policy control: parameters for admission control, policing, marking etc.

Figure 4 shows the range of potential QoS functions in the ST Ingress node and their interfaces to the rest of the BSM system, split into User, Control and Management Planes.

The figure illustrates the QoS functional blocks and message flows between functional blocks at the IP layer and above, without the satellite system-specific details. It shows the data flow in one direction only for simplicity. One Satellite Terminal (ST) connected to the user’s network acts as an ingress router to the BSM sub-network and this is connected to another ST (either a Hub ST in a star network or a User ST in a mesh network) which acts as an egress router to other networks participating in end-to-end QoS. The QoS functions of the BSM network are centralised as far as possible through a client-server model in order to coordinate the control of performance for the whole system. An Application Layer function (Service Control Function) is shown as a means to trigger QoS requests, as an alternative to network-layer signalling.

III. BSM RESOURCE MANAGEMENT

The approach adopted for the BSM QoS architecture (figures 3 & 4 show the Manager and ST parts of the architecture) is based on centralised control and management of the BSM subnetwork through a server entity called the BSM QoS Manager (BQM). The ST’s, as network edge devices, are responsible for traffic processing and policy enforcement at the ingress and egress, but they should be controlled from the BQM. The BQM should contain all the necessary functions to manage QoS for all layers above the SISAP in both Management and Control Planes. The BQM interacts with equivalent local functions in the ST’s.

The control and management functions below the SISAP (for connection control, bearer set up, BSM QoS etc.) are usually also centralised in the NCC (Network Control Centre - for the SD layers), which may be closely associated with the BQM.

Many of the functions in the BQM are standardised functions such as those for signalling (RSVP/NSIS or SIP Proxy/SDP, but others specific to the BSM, such as those for managing the BSM’s global IP and SIAF layer resources, are allocated to a functional entity called the BSM Resource Controller (BRC).

If the BSM is an independent administrative domain then the BQM acts as a self-standing PDP (Policy Decision Point). If the BSM is part of a wider domain then the BQM may act as a Local PDP (LPDP) to which the policy database is downloaded from the primary PDP, in order to avoid scalability problems and excess management traffic in making many policy requests to a PDP located outside of the BSM. The BQM PDP makes policy decisions using service-based policy rules, and communicates these decisions to the Policy Enforcement Points (PEPs) in the Resource Controllers (client) of the ST’s (STRCs).

A. BSM Resource Controller

The BRC is intended to act as a server function for the local Resource Controllers in the ST’s (STRCs) acting as clients.

The BSM Resource Controller should:

- have a view on available network resources
- have a view on the allocation of IP addresses
- have a view on the use of network resources
- control admission of new IP flows based on resource availability

- control policing functions of ST’s to allow traffic to use reserved resources.

The BRC may perform several functions such as policy control and admission control. The BRC should choose the local policy to be applied to a request from the SCF (or from other triggers) based on, for example, the application type, requested priority level, and the indicated resource reservation service class.

The BRC contains the Bandwidth Broker (BB) which (re-)negotiates SLAs to modify resources between domains. The BRC may also contain a PDP (if not included within the BB) which sets policy in PEPs or makes policy decisions on request from PEPs in the ST Resource Controllers and returns the results in the form of policy. It should also maintain a number of management information bases (MIB) for the purpose of QoS control and management of the BSM domain, for example:

- topology information base (information that the BRC uses for route selection and other management and operation purposes)
- policy information base (contains policies and other administrative regulations of the domain)
- link QoS state information base

B. *QoS Architecture Requirements - Conclusion*

The BSM QoS Architecture should support the following functional requirements:

1) Compatibility with NGN principles of uncoupling of services and networks
2) Capability to include as many of the QoS functional building blocks as needed to achieve the required overall network performance i.e. modular concept.
3) Capability to implement the range of methods for QoS signalling (request, initiation and negotiation) including:

- Service Provider-oriented model (Proxied QoS with policy-push)
- User-oriented model (User-requested QoS with policy-pull).

IV. BSM QOS FUNCTIONAL ARCHITECTURE DEFINITION

No standardised or common approach to network architecture for end-to-end QoS provision exists at present. Therefore an example of a QoS functional architecture is proposed for the BSM system which can be applied to several types of overall implementation (figures 3 & 4).

A. *QoS Interfaces to BSM Satellite-Dependent Layers*

Central to the QoS capability of the BSM is the interface of the IP layer with the lower SD layers at the SISAP. To abstract the User Plane QoS interface at the SISAP the concept of QID’s (Queue Identifiers) has been introduced [2]. These represent abstract queues available at the SISAP, each with a defined class of service for transfer of IP packets to the SD layers.

The satellite dependent lower layers are responsible for assigning satellite capacity to these abstract queues according to the specified queue properties (e.g. QoS). The QID is not limited to a capacity allocation class; it relates also to forwarding behaviour with defined properties.

A QID is only required for submitting (sending) data via the SISAP and the QID is assigned when the associated queue is opened. An open queue is uniquely identified by the associated QID: in particular, the QID is used to label all subsequent data transfers via that queue.

The way in which QIDs are mapped to the IP layer queues is an important consideration for overall QoS.

1) QID Management

An ST QID Resource Manager (STQRM) functional entity is required to manage QIDs and their mapping to BSM IDs and to the IP layer. This entity should be logically situated in both the SIAF and SDAF protocol layers, and can be part of the BSM Management Plane or Control Plane functions (ST Resource Controller).

This STQRM should also be considered as a client function to a centralised QID manager (Server) (BSMQRM) situated in the BSM QoS Manager. The BQM manages the QID resources of the BSM system and allocates QIDs and their mapping via the ST clients.

The STRC should be aware of the QID resources available at any time, either by being informed by or interrogating the BRC at IP system level (e.g. in the case of static QIDs), or by the STQRM at local SISAP level (e.g. for dynamic QIDs).

QID resources could be requested and allocated dynamically by two main paths:

1) by request through the IP layer (BRC client to BRC server by direct interaction, then forwarded to the BSM-QRM server). This latter then allocates QIDs which are distributed to the ST-QRM). The loop is closed by the BRC coordinating these resources with the NCC.
2) by request through the SD layer (BRC client to SD Resource Manager request via the control plane, followed by NCC to BSM-QRM request. This latter then allocates QIDs which are distributed to the ST-QRM). The loop is closed by coordinating these resources with the BRC for the IP layer resources and policy.

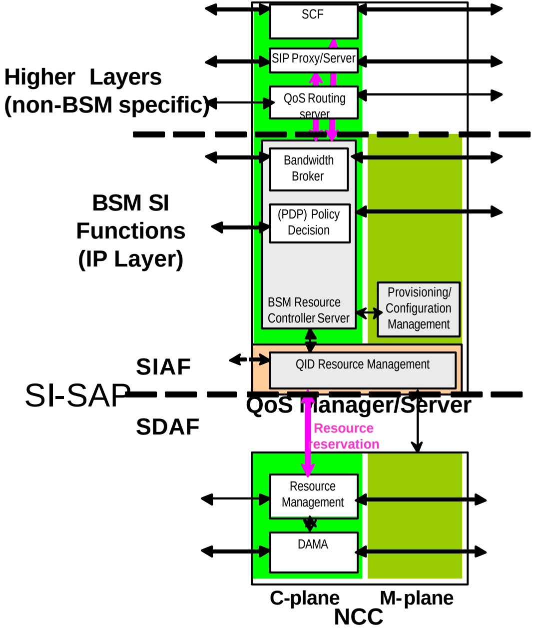

Figure 3: QoS Manager architecture across the SISAP

B. *BSM SISAP*

The SISAP represents the interface between Satellite-Independent higher layer protocols and lower layer Satellite-dependent protocols.

The different cases of interaction between QoS requests and the BSM involve not only the User Plane containing the QIDs, but also the Control and Management Planes that influence the way the QIDs are used. The interaction between the IP layer QoS and the SD layer QoS takes place across the SISAP and is thus the major issue for the BSM.

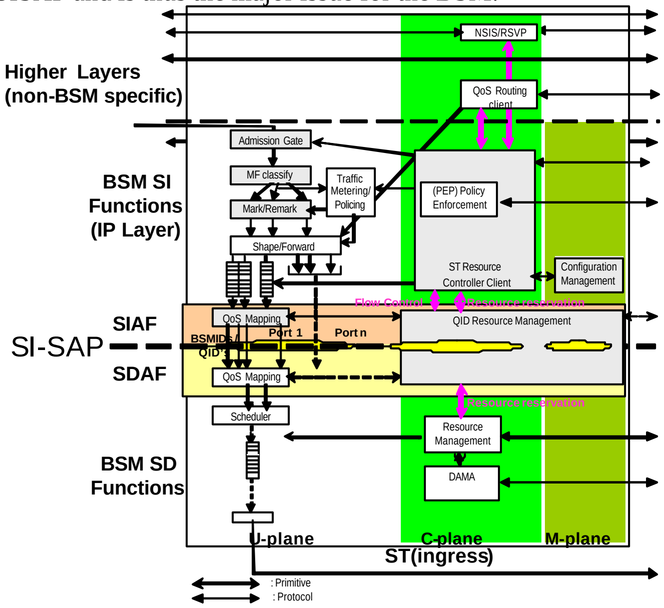

Figure 4: ST ingress architecture across the SISAP

V. DETAILED ARCHITECTURE AT THE SISAP

For simplicity, this section describes the architecture for the most complete set of functions of cases Simpler cases can be derived by omitting functions that are not required.

A. *User Plane*

In mesh networks the ST is the sole type of border node to the BSM and it must provide all required QoS functions. The asymmetric nature of star networks on the other hand means there are two types of border node: the single Hub station providing all ingress QoS processing from the external network, and its associated ST's.

1) *ST Architecture*

In the User Plane the ST must implement all traffic processing and policing functions according to a packet's class of service.

An ST User Plane QoS architecture is illustrated in Figure 5. For clarity in this figure it is assumed that there is one output port (no routing, one set of QIDs) and one subscriber per ST, either an individual or an organisation, so that one SLA and one set of IP queues is sufficient for illustration, instead of a set for each subscriber.

The ST architecture we are concerned with is based on two layers of queues: one at the IP layer and another at the SISAP identified by QIDs (further queues may exist at lower or higher layers).

The relationship between these sets of queues at the two layers is an important consideration.

Queuing could be managed solely at the IP layer, especially for relative QoS, where policies implemented would depend on available rate from the SISAP. If there are, however, different classes of service available at the SISAP, such as guaranteed rate and on-demand rate, then it would be beneficial, especially for guaranteed QoS, to map IP queues to the corresponding QIDs, in order to ensure that IP QoS parameters (such as guaranteed rate) are maintained at lower layers. The behaviour of the concatenated queues must be carefully taken into account in the overall QoS behaviour; the IP queue behaviour should not be degraded by QIDs (e.g. by increased jitter, delay, loss, etc.), and QIDs should ideally offer a transparent medium.

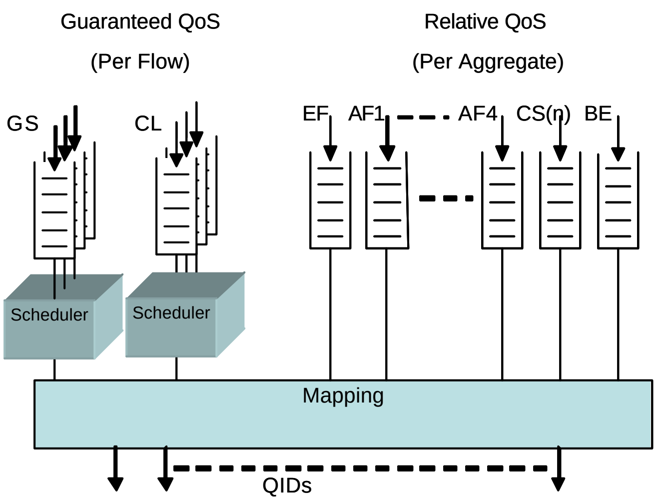

Figure 5: ST Queues for Guaranteed and Relative QoS

2) *IP Queues*

a) *Relative QoS*

The ST queuing model should support a typical DiffServ IP queue set for the BE, CS, AF, EF Per Hop Behaviours.

Other queues may be implemented depending on the PHB's defined for the BSM. A DSCP value, allocated to packets by the classifier at the ST ingress, corresponds to each queue. The queues handle aggregate IP flows of the same behaviour and control of the aggregates is typically stateless, though at the network edge admission control may be implemented.

Managing the queues includes setting the drop precedence and handling packet dropping, according to a set of rules established for each IP queue, consistent with and as an enforcement of the Traffic Conditioning Agreement (TCA) established between subscriber and the service provider (SP) as part of the SLA. This SLA includes in general support for all DSCPs. It also contains the details of the TCA for each DSCP. The TCA contains information on how metering, marking, discarding and shaping of packets must be done in order to fulfil the SLA. The TCA/SLA need to be configured in the ST. This can be done statically or dynamically; COPS protocols could be used for dynamic configuration, as per the QoS overall architecture, but other protocols (such as SNMP) may be considered as interim solutions.

b) *Guaranteed QoS*

For support of guaranteed QoS using the IntServ model, additional queues for the Guaranteed Service (GS) and Controlled Load (CL) classes are introduced with higher priority than the DiffServ queues.

Unlike DiffServ, each IP flow needs a separate queue to be maintained, as well as a separate set of all the necessary mechanisms for traffic policing etc.. Therefore state for each flow must be maintained in the ST. The parameters for each flow are established dynamically by RSVP (or NSIS) signalling, or may be established quasi-statically by network management.

These queues for each of these two guaranteed classes should be served in a round-robin manner in order that any bandwidth not used by one flow is automatically allocated to other flows and each of the queues are served fairly. Fair

Queuing provides to each flow a guaranteed minimum share of bandwidth together with the possibility to obtain more bandwidth. A scheduler is therefore needed at the output of each queue as shown in .

3) QIDs

QIDs represent abstract queues which are associated with BSM classes of service. However more QIDs than BSM traffic classes can be assigned (e.g. with different priority within a class) in order to give increased differentiation of BSM QoS.

Dimensioning the maximum SD queue size associated with a QID is an important factor in the overall QoS. The SD queue should be dimensioned to take into account any lower layer (BoD) scheduling latency. A good practice is to dimension the SD queue to accommodate the traffic accumulated in the scheduling interval at the maximum link rate or at the user-ST interface rate, whichever is higher. This is needed in order to deal with the situation immediately after a session request acceptance, and before bandwidth resources are allocated to the link after a dynamic request

In general BE SD queues can be larger than for other classes to avoid loss, as there is no committed rate and the packets can wait for longer before capacity becomes available.

4) Mapping between IP queues and QIDs

The mapping of IP queues to QIDs can in principle be flexible, without being constrained to a one-to-one relationship. If less than one QID is used per IP queue in aggregate (i.e. shared QIDs), then a scheduler is needed to differentiate between priority of IP queues and ensure fair access to QIDs.

For example within the AF class, a scheduler algorithm could determine the service discipline for packets in all AF queues into one or more QIDs. The schedulers required are therefore dependent on the mapping between IP queues and QIDs.

For guaranteed services the per-flow queues for each class need to be scheduled before merging into an aggregate for which a single QID could be assigned.

The number and characteristics of QIDs assigned must therefore also be interrelated to the mapping and scheduling configuration between these sets of queues.

B. Hub Station Architecture

Compared with typical ST's, the ingress node of the Hub Station version of an ST handles the traffic destined to many other ST's, and may be connected to several ISP networks. In many respects the Hub Station can be similar to an ST used in a mesh network, though on a different scale.

IP flows are classified and processed at the ingress for each port, according to the SLA and policy applying to each ISP. The way in which users' SLA's (e.g. between users and potential ISP's, or between users and the BSM network operator) are aggregated into the SLA's between the BSM network operator and the ISP's needs to be carefully considered.

Classification should be into a common set of BSM PHB's so that after processing and forwarding the flows can be merged and queued in one set of queues for each output port.

C. Control Plane

When an IP layer resource request is received at the ST (e.g. from RSVP or NSIS signalling etc.) the BRC client in the ST should issue a resource request to the BRC server. This message can be sent via an IP layer flow with appropriate traffic class, or possibly via a SISAP control plane primitive dedicated to IP layer signalling flows. Several mechanisms can be envisaged for subsequent actions, for example:

a) The BRC could allocate IP layer resources if there are sufficient QID resources, and reply to the ST at the IP layer.

b) if there are insufficient QID resources either:

- The BRC could immediately issue a request for SD resources across the SISAP,. This could be done via a message passed directly between the BRC Server and the NCC. If the request is accepted by the NCC, the BRC and the NCC can configure resources at the ST directly at their respective layers.
- Alternatively the BRC server could reply to the BRC client to authorise it to request SD resources across its SISAP and in the case of acceptance the STRC would be informed across the ST SISAP. The BRC would be informed by the NCC.

This SISAP layer request could be of several types according to the QoS parameters of QIDs available such as bit rate, traffic class (e.g. guaranteed rate, volume-based, etc.), delay, jitter. A QID resource request could be for new resources or for modification of existing resources. If the request is granted then the QID parameters must be passed to the BRC client, together with the updated mapping requirement.

VI. CONCLUSION

A QoS functional architecture for BSM systems has been described. More details are available in draft document TS 102462 at: http://portal.etsi.org/STFs/SES/STF283.asp.

The BSM QoS Architecture should support the following functional requirements:

1) Compatibility with NGN principles of uncoupling of services and networks
2) Capability to include as many of the QoS functional building blocks as needed to achieve the required overall network performance i.e. modular concept.
3) Capability to implement a range of methods for QoS signalling

Work is continuing on the details of this architecture within the ETSI BSM group.

ACKNOWLEDGEMENT

This work was sponsored by the European Telecommunications Standards Institute (ETSI) [3] and supported by the European Commission [9].

REFERENCES

[1] ETSI TR 101 985: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; IP over Satellite".
[2] ETSI TS 102 357 "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; Common air interface specification: Satellite Independent Service Access Point (SI-SAP)"
[3] RFC 2990 Next Steps for the IP QoS Architecture
[4] ETSI TS 185 001 "NGN; QoS Framework and Requirements"
[5] ITU-T Recommendation Y.1291 An architectural framework for support of Quality of Service in packet networks
[6] TR-059: DSL Forum; DSL Evolution - Architecture Requirements for the Support of QoS-Enabled IP Services
[7] 3GPP TS 23.107 Quality of Service (QoS) concept and architecture
[8] ETSI home page: http://portal.etsi.org/Portal_Common/home.asp
[9] European Commission home page: http://www.cordis.lu/