

# ETSI TS 102 464 V1.2.1 (2015-12)



## **Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Interworking with DiffServ QoS**

---

Reference

RTS/SES-00357

---

Keywords

broadband, interworking, IP, multimedia, QoS,  
satellite

**ETSI**

650 Route des Lucioles  
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C  
Association à but non lucratif enregistrée à la  
Sous-Préfecture de Grasse (06) N° 7803/88

---

**Important notice**

The present document can be downloaded from:  
<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at  
<http://portal.etsi.org/tb/status/status.asp>

If you find errors in the present document, please send your comment to one of the following services:  
<https://portal.etsi.org/People/CommitteeSupportStaff.aspx>

---

**Copyright Notification**

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.  
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2015.  
All rights reserved.

**DECT™**, **PLUGTESTS™**, **UMTS™** and the ETSI logo are Trade Marks of ETSI registered for the benefit of its Members.  
**3GPP™** and **LTE™** are Trade Marks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.  
**GSM®** and the GSM logo are Trade Marks registered and owned by the GSM Association.

# Contents

Intellectual Property Rights .....	5
Foreword.....	5
Modal verbs terminology.....	5
Introduction .....	5
1 Scope.....	6
2 References .....	6
2.1 Normative references .....	6
2.2 Informative references.....	7
3 Definitions and abbreviations.....	8
3.1 Definitions.....	8
3.2 Abbreviations .....	10
4 Overview .....	11
5 DiffServ BSM Functional Architecture.....	12
5.0 Aim and Scope .....	12
5.1 Overall BSM DiffServ Architecture.....	12
5.2 ST DiffServ Architecture .....	14
5.2.0 Overview .....	14
5.2.1 DiffServ Functions.....	15
5.2.1.0 Introduction.....	15
5.2.1.1 User-Plane Functions .....	15
5.2.1.2 Control-Plane Functions.....	16
5.2.2 Detailed Functional Architecture.....	17
5.2.2.0 U- and C-plane operations .....	17
5.2.2.1 Ingress ST DiffServ Functions.....	19
5.2.2.1.0 Detailed Functional Architecture.....	19
5.2.2.1.1 RSVP.....	20
5.2.2.1.2 NSIS .....	21
5.2.2.2 Egress ST DiffServ Functions.....	21
6 DiffServ QID Management .....	21
6.0 Aim and Scope .....	21
6.1 Description .....	21
6.2 QID Relationship to SD Resource Reservation.....	22
6.2.0 QID-to-SD Mapping .....	22
6.2.1 Static SD Resources.....	23
6.2.1.0 Overview.....	23
6.2.1.1 Static QIDs.....	23
6.2.1.2 Dynamic QIDs .....	23
6.2.2 Dynamic SD Resources .....	24
6.2.2.0 Overview.....	24
6.2.2.1 Static QIDs.....	25
6.2.2.2 Dynamic QIDs .....	25
6.2.3 Summary.....	26
6.3 QID Relationship to SI layers.....	26
6.3.0 DiffServ Per Hop Behaviours (PHBs) .....	26
6.3.1 Best Effort.....	27
6.3.2 Expedited Forwarding.....	28
6.3.3 Assured Forwarding.....	29
6.3.4 Class Selector.....	30
6.3.5 Summary.....	31
<b>Annex A (informative): Relationship to the RSM-B Air Interface family (DVB-RCS).....</b>	<b>32</b>
A.0 General .....	32

A.1	SATLIFE satellite system summary.....	32
A.2	SATLIFE QoS model.....	33
A.3	RCST/NCC Model .....	35
A.3.0	Overview .....	35
A.3.1	User Plane .....	35
A.3.2	Control Plane.....	36
<b>Annex B (informative):</b>	<b>Relationship to the TSS-A Air Interface Family (DVB-RCS) .....</b>	<b>37</b>
<b>Annex C (informative):</b>	<b>DiffServ DSCPs.....</b>	<b>38</b>
<b>Annex D (informative):</b>	<b>Bibliography.....</b>	<b>39</b>
History .....		40

---

## Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<http://ipr.etsi.org>).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

---

## Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Satellite Earth Stations and Systems (SES).

---

## Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

---

## Introduction

The BSM architecture is characterized by the separation between common Satellite-Independent (SI) protocol layers and alternative lower Satellite-Dependent (SD) layers, which are connected through the Satellite Independent Service Access Point (SI-SAP) [1]. The general issues concerning the architecture of BSM systems are described in ETSI TR 101 984 [i.1], further specific requirements and functional models for Quality of Service (QoS) concerning IP-over-satellite aspects are presented in ETSI TR 101 985 [i.2] and ETSI TR 102 157 [i.3].

In general the SI-SAP offers an agnostic interface to whichever SD layer is used. So QoS provision in the BSM architecture has to face the issue of traversing the SI-SAP interface by means of standardized signalling, which is expected to enable on one side maintaining compatibility with existing QoS functions required in the IP layers and above, and on the other side communication to the lower layer entities deputed to QoS accomplishment.

At the IP layer, two principal techniques for QoS provision exist: DiffServ [7], and RSVP/IntServ [4], [5]. At the SD layers more sophisticated QoS methods are closely linked to lower layer resource management and control, they strongly depend on the satellite technology adopted and on the particular implementation.

For QoS provision in a BSM network the concept of QIDs (Queue Identifiers) is a key concept [2]. They represent abstract queues, 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 and/or particular forwarding behaviour to these abstract queues according to defined properties. The reader should in particular refer to ETSI TS 102 463 [i.14], for a detailed description of QIDs and of the associated primitives.

---

# 1 Scope

The aim of the present document is to define an open specification for enabling QoS for IP-based multimedia satellite systems, based on the DiffServ model. If IP packets entering the BSM network require a particular QoS treatment, they have to be mapped onto QIDs. The choice of the QID to be used inside the BSM network is thus particularly important. So the present document specifies the allocation of the QIDs and their mapping to IP QoS classes, when DiffServ is used to provide QoS at IP layer. The present document assumes the QoS functional architecture described in ETSI TS 102 462 [2].

The present document describes in detail how QIDs are defined, how they are allocated and handled by the BSM network, and the requirements needed by sending and receiving Satellite Terminals (STs) in a BSM network to provide QID management functionalities. The present document also defines the primitives that should be used across the SI-SAP when allocating QIDs, when mapping DiffServ Code Points (DSCPs) and IP services to QIDs, when mapping QIDs to SD queues.

Details on the QID mapping are presented with some examples. Some cases are presented to show the potential evolution from a simple QoS solution with quasi-static QID allocation to more sophisticated services with dynamic resource reservation.

The combination of DiffServ with multicast transmissions is out of scope of the present document, as well as the use of Explicit Congestion Notification (ECN), which was linked to DiffServ only for historical reasons, as the ECN bits are the two least significant bits of the IPv4 ToS octet. This is better explained in clause 4.

---

## 2 References

### 2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the reference document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at <http://docbox.etsi.org/Reference>.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

- [1] ETSI TS 102 357: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Common Air interface specification; Satellite Independent Service Access Point (SI-SAP) interface: Primitives".
- [2] ETSI TS 102 462: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); QoS Functional Architecture".
- [3] IETF RFC 3168: "The Addition of Explicit Congestion Notification (ECN) to IP", September 2001.
- [4] IETF RFC 1633: "Integrated Services in the Internet Architecture: an Overview", June 1994.
- [5] IETF RFC 2210: "The Use of RSVP with IETF Integrated Services", September 1997.
- [6] IETF RFC 2474: "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", December 1998.
- [7] IETF RFC 2475: "An Architecture for Differentiated Service", December 1998.
- [8] IETF RFC 2597: "Assured Forwarding PHB Group", June 1999.
- [9] IETF RFC 3246: "An Expedited Forwarding PHB (Per-Hop Behavior)", March 2002.

- [10] IETF RFC 3247: "Supplemental Information for the New Definition of the EF PHB (Expedited Forwarding Per-Hop Behavior)", March 2002.
- [11] IETF RFC 3260: "New Terminology and Clarifications for Diffserv", April 2002.

## 2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the reference document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

- [i.1] ETSI TR 101 984: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Services and architectures".
- [i.2] ETSI TR 101 985: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; IP over Satellite".
- [i.3] ETSI TR 102 157: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; IP Interworking over satellite; Performance, Availability and Quality of Service".
- [i.4] IETF RFC 2998: "A Framework for Integrated Services Operation over Diffserv Networks", November 2000.
- [i.5] IETF RFC 4080: "Next Steps in Signaling (NSIS): Framework", June 2005.
- [i.6] Satellite Access Technologies: Leading Improvements For Europe.

NOTE: Available at <http://www.satlife.org>

- [i.7] Wittig, Manfred; Casas, Jose-Maria. "A communications switchboard in the sky: AmerHis". ESA Bulletin, No. 115. August 2003.
- [i.8] S. Chacón, J-L. Casas, A. Cal, R. Rey, J. Prat, A. Rodriguez, J. de la Plaza, C. Nieto, F. Ruiz Piñar. Multimedia Applications of the Integrated Broadcast Interaction System (IBIS). ESA Workshop on Digital Signal Processing, Lisbon (Portugal), October 2001.
- [i.9] ETSI TS 102 429-1: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Regenerative Satellite Mesh - B (RSM-B); DVB-S/DVB-RCS family for regenerative satellites; Part 1: System Overview".
- [i.10] "SatLabs System Recommendations v2.0", November 2006.

NOTE: Available at <http://satlabs.org/>.

- [i.11] S. Floyd and V. Jacobson "Random Early Detection Gateways for Congestion Avoidance". IEEE/ACM Transactions on Networking, Vol. 1, Issue 4, August 1993.
- [i.12] ETSI TS 102 429-3: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Regenerative Satellite Mesh - B (RSM-B); DVB-S/DVB-RCS family for regenerative satellites; Part 3: Connection control protocol".
- [i.13] ETSI TS 102 402: "Satellite Earth Station and systems (SES); Broadband Satellite Multimedia; Transparent Satellite Star - A (TSS-A); DVB-S/DVB-RCS for transparent satellites; Sub-family 1 (TSS-A1)".
- [i.14] ETSI TS 102 463: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Interworking with IntServ QoS".

## 3 Definitions and abbreviations

### 3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

**architecture:** abstract representation of a communications system

NOTE: Three complementary types of architecture are defined:

- **Functional Architecture:** The discrete functional elements of the system and the associated logical interfaces.
- **Network Architecture:** The discrete physical (network) elements of the system and the associated physical interfaces.
- **Protocol Architecture:** The protocol stacks involved in the operation of the system and the associated peering relationships.

**bearer service:** type of telecommunication service that provides the capability for the transmission of signals between user-network interfaces

**Behaviour Aggregate (BA):** collection of packets with the same DS code point crossing a link in a particular direction

**Best-Effort (BE) service:** service that offers no QoS guarantees, just end-to-end connectivity

NOTE: When using queuing to prevent congestion BE queues are always the first ones to experience packet drop.

**BSM Bearer service:** telecommunication service that a BSM subnetwork provides between a pair of SI-SAPs in different STs

**Class of Service (COS):** way to divide traffic into separate categories (classes) to provide appropriate QoS services to each class within the network

**classification:** examination of an IP packet to determine the COS to which the packet should belong

**code point:** specific value of the DSCP portion of the DS field

NOTE: Recommended code points should map to specific standardised Per-Hop Behaviours (PHBs). Multiple code points may map to the same PHB.

**connection oriented:** communication method in which communication proceeds through three well-defined phases: connection establishment, data transfer, and connection release

**connectionless:** communication method that allows the transfer of information between users without the need for connection establishment procedures

**control plane:** plane with a layered structure that performs the call control and connection control functions and deals with the signalling necessary to set up, supervise and release calls and connections

**data link layer:** second layer of the OSI model it provides connectivity between segments of the network (bridging); in addition the data link may perform session control and some configuration

**delay variation (or delay jitter):** difference in delay between successive packet arrivals (of the same flow) at the egress of the network

**Differentiated services (DiffServ):** services based on statistical (aggregated flows) guarantees and resulting in "soft" QoS

NOTE: Using packet markings (code points) and queuing policies it results in some traffic to be better treated or given priority over other (use more bandwidth, experience less loss, etc.).



**Differentiated Services Codepoint (DSCP):** value which is encoded in the DS field, and which each DS node use to select the PHB which is to be experienced by each packet it forwards

**Differentiated Services field (DS field):** six most significant bits of the (former) IPv4 TOS octet or the (former) IPv6 Traffic Class octet

**DS domain:** contiguous set of DS nodes which operate with a common service provisioning policy and set of PHB groups implemented on each node

**flow:** flow of packets is the traffic associated with a given connection or connectionless stream having the same source host, destination host, class of service, and session identification

**Integrated services (IntServ):** using RVSP this results in deterministic reservation of network resources and QoS for specific traffic and/or for specific IP flows

**marking:** to set the class of service or DSCP of a packet

**metering:** process of measuring the temporal properties (e.g. rate) of a traffic stream selected by a classifier

NOTE: The instantaneous state of this process may be used to affect the operation of shaping, or dropping.

**Network Control Centre (NCC):** equipment at OSI Layer 2 that controls the access of terminals to a satellite network, including element management and resource management functionality

**Per-Hop Behaviour (PHB):** externally observable forwarding treatment applied at a differentiated services-compliant node to a behaviour aggregate

**policing:** process of discarding packets (by a dropper) within a traffic stream in accordance with the state of a corresponding meter enforcing a traffic profile

**Quality of Service (QoS):** ability to segment traffic or differentiate between traffic types in order for the network to treat certain traffic differently from others. QoS encompasses both the service categorization and the overall performance of the network for each category.

NOTE: It also refers to the capability of a network to provide better service to selected network traffic over various technologies and IP-routed networks that may use any or all of the underlying technologies.

**QoS Parameters:** parameters that will be specified or monitored to ensure QoS

**service levels:** end-to-end QoS capabilities of the network which will enable it to deliver a service needed by a specific mix of network traffic

NOTE: The services themselves may differ in their level of QoS.

**Service Level Agreement (SLA):** agreement between a Service Provider (SP) and its subscriber (or between an SP and an access network operator), characterized by the choice of one data transfer capability and the allocation attribute related to this transfer capability

NOTE: An SLA can also include elements related to traffic policy and availability. It is agreed upon at the initiation of the contract and normally remains the same for all the contract duration.

**Traffic Conditioning Agreement (TCA):** agreement specifying classifier rules and any corresponding traffic profiles and metering, marking, discarding and/or shaping rules which are to apply to the traffic streams selected by the classifier

NOTE: A TCA encompasses all of the traffic conditioning rules explicitly specified within an SLA along with all of the rules implicit from the relevant service requirements and/or from a DS domain's service provisioning policy.

**transfer capability:** capability of transfer of information through a network

NOTE: This term can be used to characterize a telecommunications service or bearer.

**user plane:** plane with a layered structure that provides user information transfer, along with associated controls (e.g. flow control, recovery from errors, etc.)

## 3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AF	Assured Forwarding
AF/BE	Assured Forwarding – Best Effort
AQM	Active Queue Management
BA	Behavior Aggregate
BE	Best Effort
BoD	Bandwidth on Demand
BSM	Broadband Satellite Multimedia
BSM_ID	BSM IDentifier
BSMQRM	BSM QID Resource Manager
C2P	Connection Control Protocol
CBWFQ	Class Based Weighted Fair Queuing
COPS	Common Open Policy Service
CS	Class Selector
DAMA	Demand Assignment Multiple Access
DiffServ	Differentiated Services (IETF)
DS	DiffServ
DSCP	DiffServ Code Point
DVB-RCS	Digital Video Broadcasting with Return Channel via Satellite
DVB-S	Digital Video Broadcasting - Satellite
ECN	Explicit Congestion Notification
EF	Expedited Forwarding
EXP/LU	Experimental / Local Use
FCA	Free Capacity Assignment
HP	High Priority
IANA	Internet Assigned Numbers Authority
IETF	Internet Engineering Task Force
IntServ	Integrated Services (IETF)
IP	Internet Protocol
L2	Layer 2
LP	Low Priority
MAC	Medium Access Control
MF	MultiField
MPE	Multi-Protocol Encapsulation
MPEG	Moving Pictures Expert Group
MPEG2-TS	MPEG2 - Transport Stream
MS	Management Station
NCC	Network Control Centre
NSIS	Next Steps In Signalling
OSI	Open Standards Institute
PHB	Per-Hop Behavior
PID	Packet IDentifier
QID	Queue IDentifier
QIDSPEC	QID QoS SPECifications
QoS	Quality of Service
RBDC	Rate Based Dynamic Capacity
RC	Request Class
RCST	DVB-RCS Terminal
RED	Random Early Detection
RFC	Request For Comments
RSGW	Regenerative Satellite GateWays
RSM-B	Regenerative Satellite Mesh-B
RSVP	Resource ReserVation Protocol
SD	Satellite Dependent
SDAF	Satellite Dependent Adaptation Functions
SI	Satellite Independent
SIAF	Satellite Independent Adaptation Functions
SI-SAP	Satellite Independent-Service Access Point

SLA	Service Level Agreement
SNMP	Simple Network Management Protocol
SP	Service Provider
ST	Satellite Terminal
STQRM	ST QID Resource Manager
TCA	Traffic Conditioning Agreement
TCP	Transmission Control Protocol
ToS	Type of Service
TSS-A	Transparent Satellite Star-A
UDP	User Datagram Protocol
VBDC	Volume Based Dynamic Capacity
VCI	Virtual Circuit Identifier
VPI	Virtual Path Identifier
WFQ	Weighted Fair Queuing
WRED	Weighted Random Early Detection

---

## 4 Overview

DiffServ is an IETF solution to provide "scalable service differentiation on the Internet" [7]. Its main idea is to aggregate IP flows by means of IP-layer packet marking using the DSCP (DiffServ code point) field of the IP header, and to provide the aggregate with the same QoS level.

So within DiffServ domains, QoS treatments are only provided based on the DSCP marked in each packet's IP header. In order to obtain a particular level of service, it is necessary to mark packet headers with the correct DSCP when the packets enter the DS (DiffServ) domain. The correct DSCP is determined by pre-defined policies and SLAs, or optionally set dynamically by means of signalling protocols (such as NSIS, RSVP, etc.). The IP packets (or flows) marked with the same DSCP constitute a Behavior Aggregate (BA), and they receive from a DiffServ node the same level of service, i.e. the same Per-Hop Behavior (PHB).

If the BSM system belongs to a DiffServ domain, PHBs have to be translated into QoS treatments that the packets receive from one particular ST when they are forwarded over the satellite. So DSCPs need to be carefully mapped to SD classes of service. Since SD classes are system dependent, it is necessary to abstract from the lower layers and to provide the SI layers with a common and agnostic interface. For this reason central to the DiffServ capability, and more in general to the QoS capability of BSM systems, is the concept of QIDs (Queue Identifiers) as outlined in ETSI TS 102 462 [2] and ETSI TS 102 463 [i.14]).

QIDs represent the abstract queues available at the SI-SAP, they should be seen as an association between IP queues used by DiffServ and layer-2 (L2) queues. Each QID offers a defined class of service for transfer of IP packets to the SD layers. Thus QID are abstract queues in the sense that they can be seen as a mapping between IP and SD queues, but they have real properties in the sense that they represent to the SI layers the real properties of the SD queues.

The satellite dependent lower layers are responsible for assigning satellite capacity to these abstract queues according to the specified queue properties. The QID is not limited to a capacity allocation class, it relates also to forwarding behaviour with defined properties. All these QID properties have an impact into the properties of the IP DiffServ queues which the QID is associated to.

The BSM system will most likely be only a portion of the overall path traversed by packets which require end-to-end QoS. The transmission from an ST to the satellite and down to another ST (or hub) is considered, at IP layer, as one single hop, or at maximum as two hops, in case of a BSM mesh network topology where routing is performed on-board the satellite. In a BSM system the single IP hop is shared among many STs, and this makes the QoS problem challenging.

The present document will analyze the way to provide QoS only inside the BSM system, i.e. over one single IP hop, by means of the DiffServ paradigm, and the most appropriate way to handle the SD resources. So the provision of end-to-end QoS is out of the scope of the present document. It is assumed that the QoS treatment expected by each particular IP flow or packet from the BSM is known to the BSM system by means of external information or signalling (e.g. SLA). The BSM system will take this information into account in the SD resource management.

The satellite hop remains one single IP hop also in case of broadcast or multicast. This is a very important aspect which makes the use of DiffServ over BSM system very powerful, and very simple with respect to terrestrial QoS-aware multicasting, where the service level has to be monitored over a multicast tree, and thus along different paths. Nevertheless DiffServ-aware multicast transmissions are out of scope of the present document.

For the sake of clarity, the following IETF definitions shall apply to the present document [11]:

- the Differentiated Services field (DS field) is the six most significant bits of the (former) IPv4 TOS octet or the (former) IPv6 Traffic Class octet;
- the Differentiated Services Codepoint (DSCP) is a value which is encoded in the DS field, and which each DS node should use to select the PHB which is to be experienced by each packet it forwards.

Thus the two least significant bits of the IPv4 TOS octet and the IPv6 Traffic Class octet are not used by DiffServ. They have been assigned for use of Explicit Congestion Notification (ECN) [3] and so their use is out of scope of the present document. DSCP markings may be used in the future to modify (or signal alternative semantics) for the operation of ECN.

---

## 5 DiffServ BSM Functional Architecture

### 5.0 Aim and Scope

This clause clarifies the entities and functionalities involved in the QoS management process, when the DiffServ paradigm is adopted. This DiffServ functional architecture is compliant with the one presented in ETSI TS 102 462 [2].

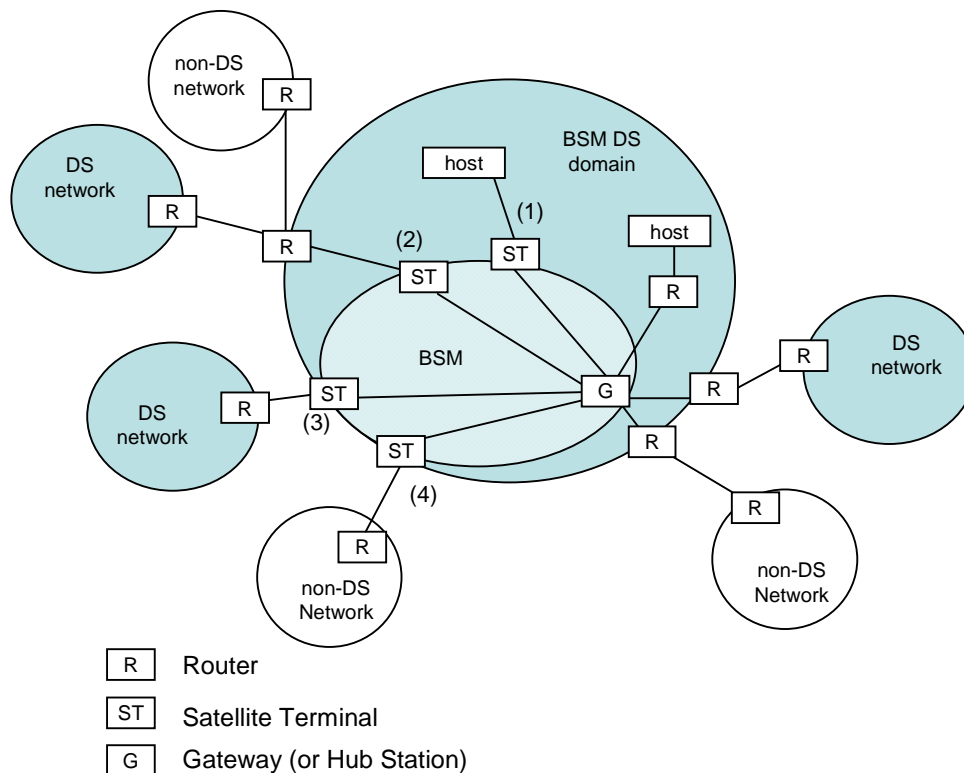
### 5.1 Overall BSM DiffServ Architecture

A BSM system may constitute part of a DiffServ domain, compliant with the architecture presented in IETF RFC 2475 [7]. The domain is referred to as the BSM DS domain. The domain may extend beyond the STs, but in general the satellite will be the critical part of the domain for the management of the resources in case of QoS provision.

The STs are the edge nodes of the BSM system, but they might be or not be the edge nodes of the BSM DS domain. Figure 1 details the architecture shown in ETSI TS 102 462 [2], and highlights four possible scenarios for the BSM DS domain topology, which are relevant to this discussion:

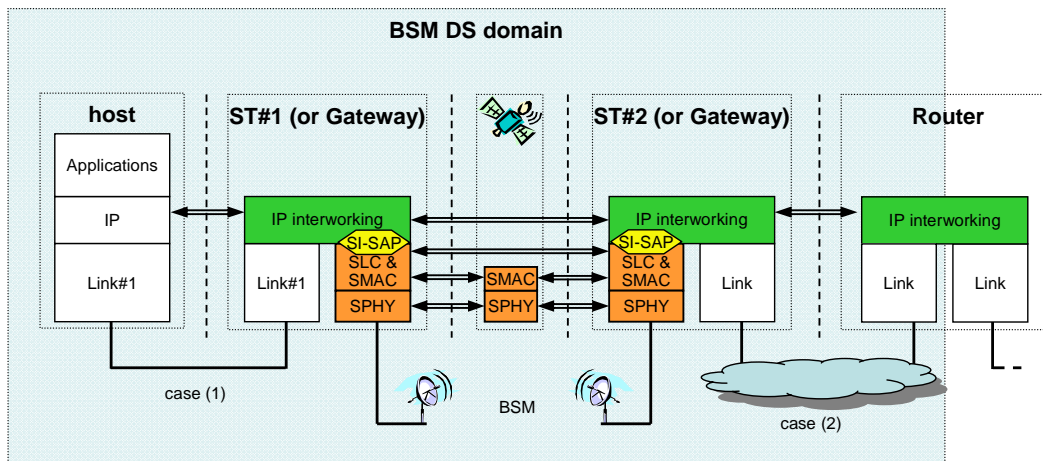
- case (1): ST directly connected to a BSM DS host;
- case (2): ST connected to a BSM DS router via one or more hops;
- case (3): ST directly connected to an external DS router;
- case (4): ST directly connected to a non-DS router.

In each of these cases the ST might need to perform different DS functions both in the user and in the control plane, this is intrinsically linked to the way DS works. In particular in a DS domain the edge routers perform a number of important operations: traffic classification, packet marking, traffic policing, traffic shaping, and, optionally, admission control and resource reservation (see ETSI TS 102 462 [2], clause 8.1, for a detailed explanation of these mechanisms). On the other hand DS core nodes only apply the appropriate PHB to IP packets based on the marked DSCP: IP packets are forwarded to the appropriate IP queue (this operation can be defined as DSCP classification). So depending on whether an ST is at the edge of the BSM DS domain or not, it might need to perform these operations or not. This will be discussed in detail for the four cases in clause 5.2.1.

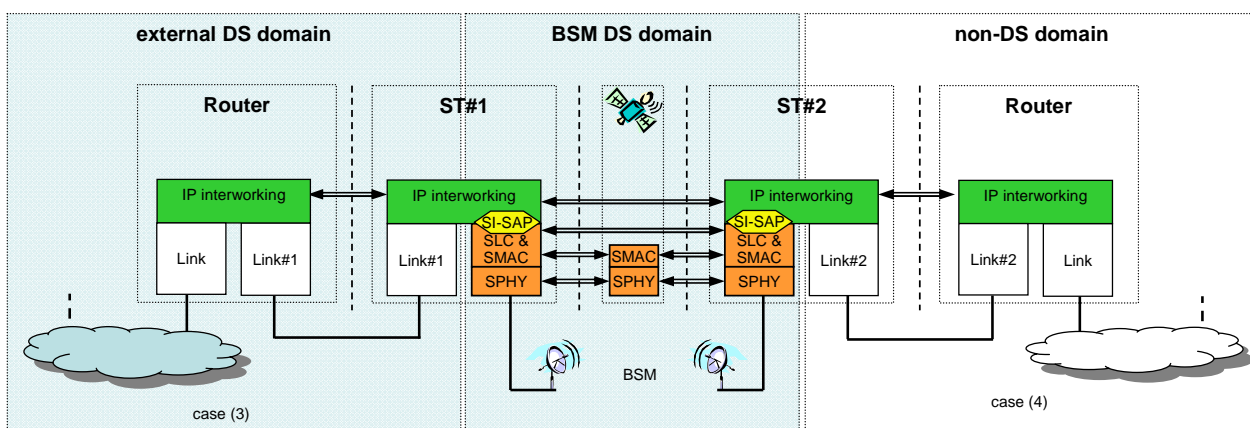


**Figure 1: Possible scenarios of the BSM DS domain topology**

Figures 2 and 3 show, for the four possible cases, the details of the ST protocol stack and its connections to external entities.



**Figure 2: ST details with respect to its location in the BSM DS domain, cases (1) and (2)**



**Figure 3: ST details with respect to its location in the BSM DS domain, cases (3) and (4)**

In general it is assumed that the satellite gateway (or the hub station) will not be an edge node of the BSM DS domain, but it will be connected to routers which perform the required DiffServ border functionalities. For this reason the gateway can only be found in cases (1) or (2) in the place of the ST. The operations to be performed by the gateway in these cases will be also analysed in the next clause 5.2.1.

## 5.2 ST DiffServ Architecture

### 5.2.0 Overview

The operations which might be performed by a DiffServ node are: traffic classification, packet marking, traffic policing, traffic shaping, DSCP classification, and, optionally, admission control and resource reservation.

Traffic classification, packet marking, traffic policing, traffic shaping, and DSCP classification can be considered user-plane mechanisms. The remaining functions (admission control and resource reservation) can be seen as control-plane operations. The relationship of these functionalities to the general protocol architecture of a DiffServ ST is shown in figure 4.

As it can be seen in the figure, all user-plane functions are located above the SI-SAP, they are functions which are present in all IP DiffServ nodes. On the other hand the control-plane functions are distributed across the SI-SAP and require interactions of SI and SD layers.

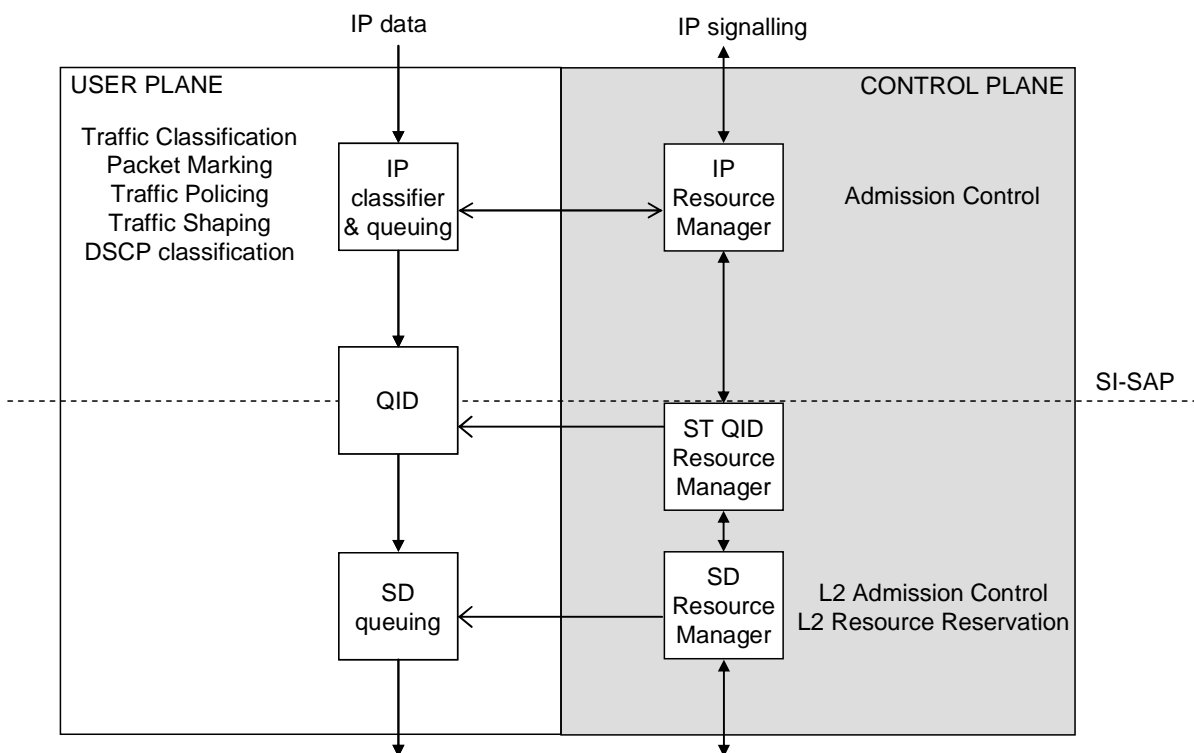


Figure 4: DiffServ-aware ST functional relationship to protocol layers and planes

## 5.2.1 DiffServ Functions

### 5.2.1.0 Introduction

In this clause the functions required in a DiffServ-aware ST are analyzed for each of the four scenarios presented in clause 5.1. It should be noted that an ST (or more likely a gateway) may have multiple terrestrial interfaces, beyond the satellite interface to the BSM system. On each one of the terrestrial interfaces the connection may represent one of the four mentioned cases. So there might be the need for a single ST (or a gateway) to handle multiple scenarios simultaneously. Since the different interfaces can be treated independently, this is not a problem, this can be realized by applying the appropriate behaviour in each case, according to the indications given in the two following clauses 5.2.1.1 and 5.2.1.2.

#### 5.2.1.1 User-Plane Functions

In the case the ST is directly connected to a host inside the BSM DS domain (see case (1) in figure 2), it is assumed that the host operating system correctly marks the DSCP in the transmitted packets. This operation is commonly called "host marking", and it can be done according to static or dynamic rules. Thus in this case the ST relies on the host for the operations of traffic classification and packet marking. On the other hand the ST normally cannot rely on the host for the purpose of traffic policing, and traffic shaping (if needed), so these functions shall be performed by the ST itself. Case (1) is plausible in some specific scenarios, and, for the sake of completeness, it is addressed in this clause, but in common practice end hosts should not be trusted when setting DSCPs.

In case (2) the ST is connected to another device in the same DS domain, which is not a host or an end system (see case (2) in figure 2). In this case it is assumed that one edge router of the BSM DS domain takes care of all mentioned user-plane functions. This applies either if the edge router is connected to another DS network and if it is connected to a non-DS one. Of course the edge router might not be directly connected to the ST. This router will be configured to identify IP flows, based on multifield (MF) classification, according to pre-defined criteria, and to consequently mark the IP headers. This can be also done either statically or dynamically.

When the ST is not performing the marking itself, but it is relying on a trusted entity for that (the edge router, case (2), or the host, case (1)), it may check the mark and change it, when necessary, but this is not mandatory: traffic classification and packet re-marking are optional in these two cases for the ST. On the other hand in case (2) the ST shall completely rely on the edge entity for the operations of traffic policing and shaping.

In the cases the ST is at the border of the BSM DS domain (cases (3) and (4) in figure 3), it has to perform all the user-plane operations that are normally applied by edge routers in a DS domain. The difference between scenario (3) and (4) is the following: In case (3), packets are coming from another DS domain, so they are marked, but these markings might have a different QoS meaning; in case (4) packets are not marked, as they come from a DS-unaware network. So in case (3) re-marking has to be performed by the ST, and traffic classification might not be necessary; in case (4) traffic classification is mandatory at the ST.

The case of a DS-unaware host directly attached to an ST is included in scenario (4), while the case of a host (with no router capabilities) belonging to another DS domain directly attached to an ST, is the same as (3).

In all scenarios DSCP classification is required, as it is the core functionality of DiffServ.

Table 1 summarizes the user-plane functions which shall be performed by the ST in the different scenarios, and the ones which might be needed (optional).

**Table 1: DiffServ User-Plane functions to be implemented in the ingress ST**

Scenarios	User-Plane Functions				
	Traffic Classification	Packet (Re-)Marking	Traffic Policing	Traffic Shaping	DSCP Classification
<b>(1): ST connects to BSM DS host</b>	O	O	M	O	M
<b>(2): ST connects to BSM DS router</b>	O	O			M
<b>(3): ST connects to external DS router</b>	O	M	M	O	M
<b>(4): ST connects to non-DS router</b>	M	M	M	O	M
NOTE:	M: mandatory function O: optional function empty cell: function not required				

Normally the satellite gateway (or the hub) will not be directly connected to an external router or host, but to another entity in the BSM DS domain. So the functions to be implemented in the gateway (or hub) in these cases can be derived from scenarios (1) and (2).

### 5.2.1.2 Control-Plane Functions

In terrestrial DiffServ systems the management of the resources is commonly (even if not always) done at IP layer, as it is very much related to IP-layer parameters, such as routing and traffic characteristics (e.g. bit rate). Links normally have a fixed capacity, and the utilization of that capacity only depends on how the IP flows are routed in the domain and on the capacity consumed by each flow.

On the other hand in a BSM system the management of layer-2 resources acquires a much higher importance, and thus the control-plane functions require interaction of the SI and SD layers.

So admission control and resource reservation, which are optional in common terrestrial DiffServ domains, might need to be implemented in BSM systems, in particular at SD layers. In a DiffServ-aware ST resource reservation may remain optional at IP layer, but it normally needs to be performed at SD layers in any case, even if QoS differentiation is not guaranteed. Similar idea applies to admission control, but this function remains optional also at SD layer if the IP layer is not expecting a notification on the admission of new IP flows (which is common for traditional DiffServ).



**Table 2: DiffServ Control-Plane functions to be implemented in the ingress ST**

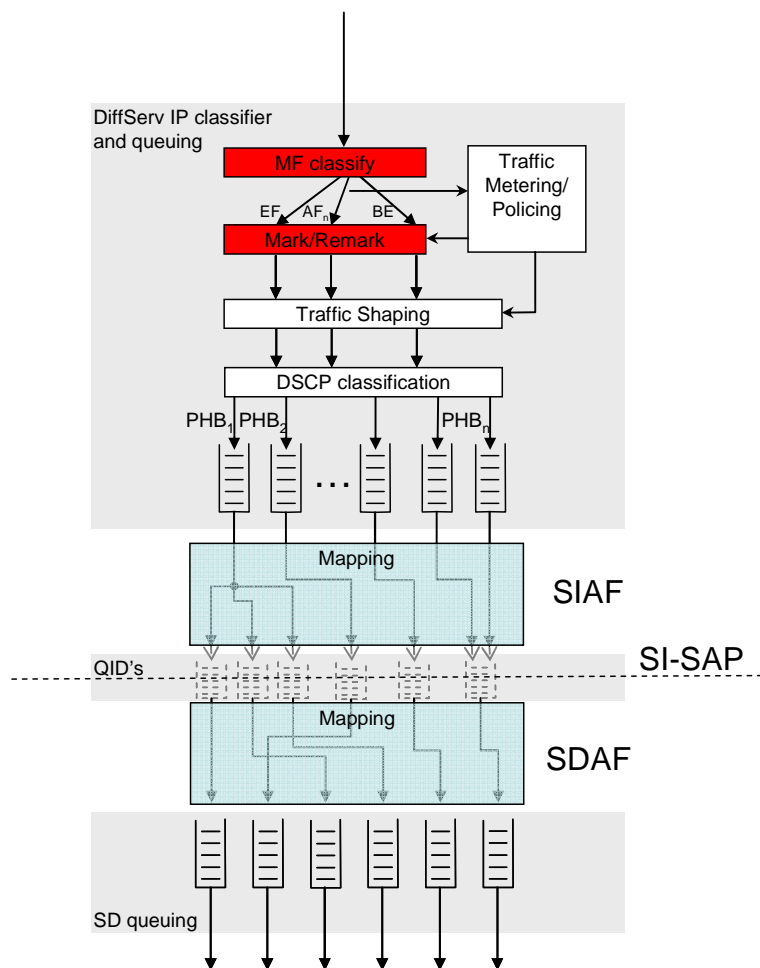
Scenarios	Control-Plane Functions	
	Admission Control	Resource Reservation
(1): ST connects to BSM DS host	O	M
(2): ST connects to BSM DS router	O	M
(3): ST connects to external DS router	O	M
(4): ST connects to non-DS router	O	M

## 5.2.2 Detailed Functional Architecture

### 5.2.2.0 U- and C-plane operations

DiffServ implementation in BSM is mainly connected to the relation between IP queues and SD queues. This relation is mediated in a BSM by the concept of QID. The introduction of QID is needed to abstract from real layer-2 implementation, as SD queues can be of many and different types.

Figure 5 displays the relationship between the traditional DiffServ IP queue management, which is performed above the SI-SAP, and the lower layers queues (SD queues and QIDs). The figure shows the flow of data over the different IP, SD queues, and QIDs, in the user-plane, and the two stages of mapping. In this figure QIDs are depicted with dotted lines, as they represent abstract queues. The mapping can make use of schedulers in real implementations. A link-layer scheduler may be also used below the SD queues (it is not shown in figure 5).



**Figure 5: User-Plane detailed architecture of a DiffServ-aware ST**

QIDs are locally handled by the ST QID Resource Manager (STQRM) which is logically located in the SDAF module. The STQRM receives QID allocation requests from the IP resource manager, and sets the allocation of SD resources accordingly. The particular submodule of the IP resource manager able to interact with the STQRM will be part of the SIAF, whereas the submodule of the SD resource manager able to interact with the STQRM will be part of the SDAF.

This way of operating in the control-plane is shown in figure 6:

- 1) The IP resource manager interacts with the IP classifier and queuing module in the user-plane and with external IP signalling (optional) to understand when and whether new resources are needed (or not needed anymore). The IP DiffServ queues may remain static, but the traffic situation might change.

MESSAGES: (1a) terminal defined, (1b) standard IP signalling, e.g. RSVP, NSIS.

- 2) So if the situation changes this module should notify the STQRM and take appropriate actions to allocate/release resources at the SD layers.

MESSAGES: (2) SI-SAP primitives, see ETSI TS 102 463 [i.14].

- 3) This triggers an exchange of messages along the red lines, down to the SD layer and the NCC, since the resources at the SD layer will be, most likely, centrally managed.

MESSAGES: (3a) terminal defined, (3b) BSM defined.

- 4) The reply of the NCC will follow the green lines up to the IP resource manager.

- 5) MESSAGES: (4a) BSM defined, (4b) terminal defined, (4c) SI-SAP primitives, see ETSI TS 102 463 [i.14].

- 6) These replies will enable the (re)configuration of the queue structure and of the mapping shown in figure 5.

MESSAGES: (5a) terminal defined, (5b) terminal defined, (5c) terminal defined.

- 7) In the end these operations might trigger responses at IP signalling level (optional).

MESSAGES: (6) standard IP signalling e.g. RSVP, NSIS.

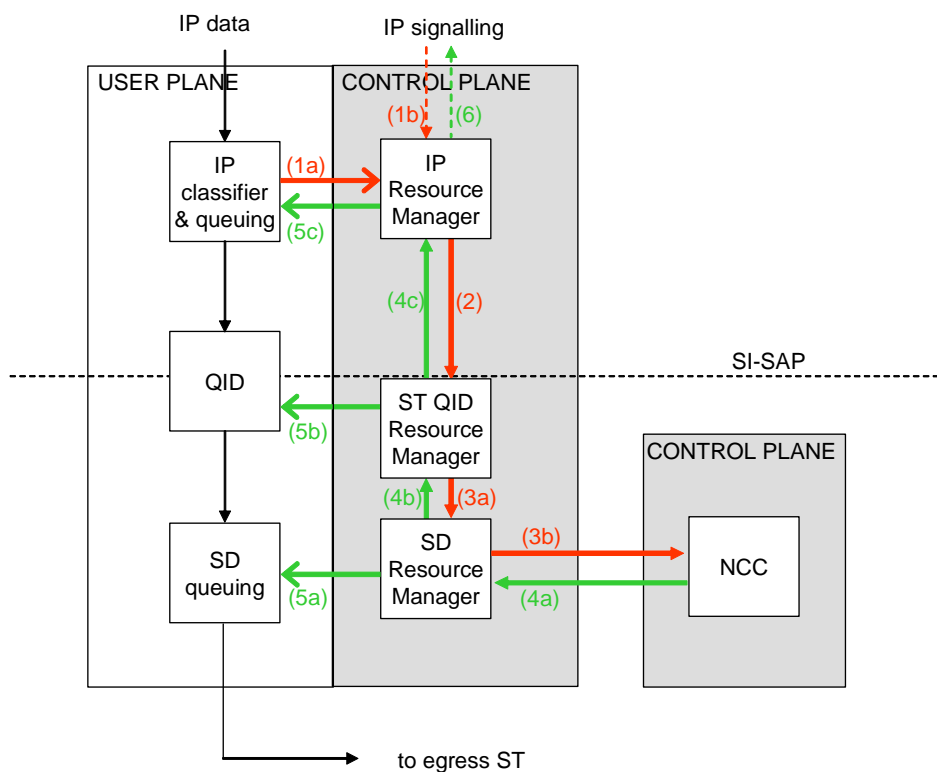


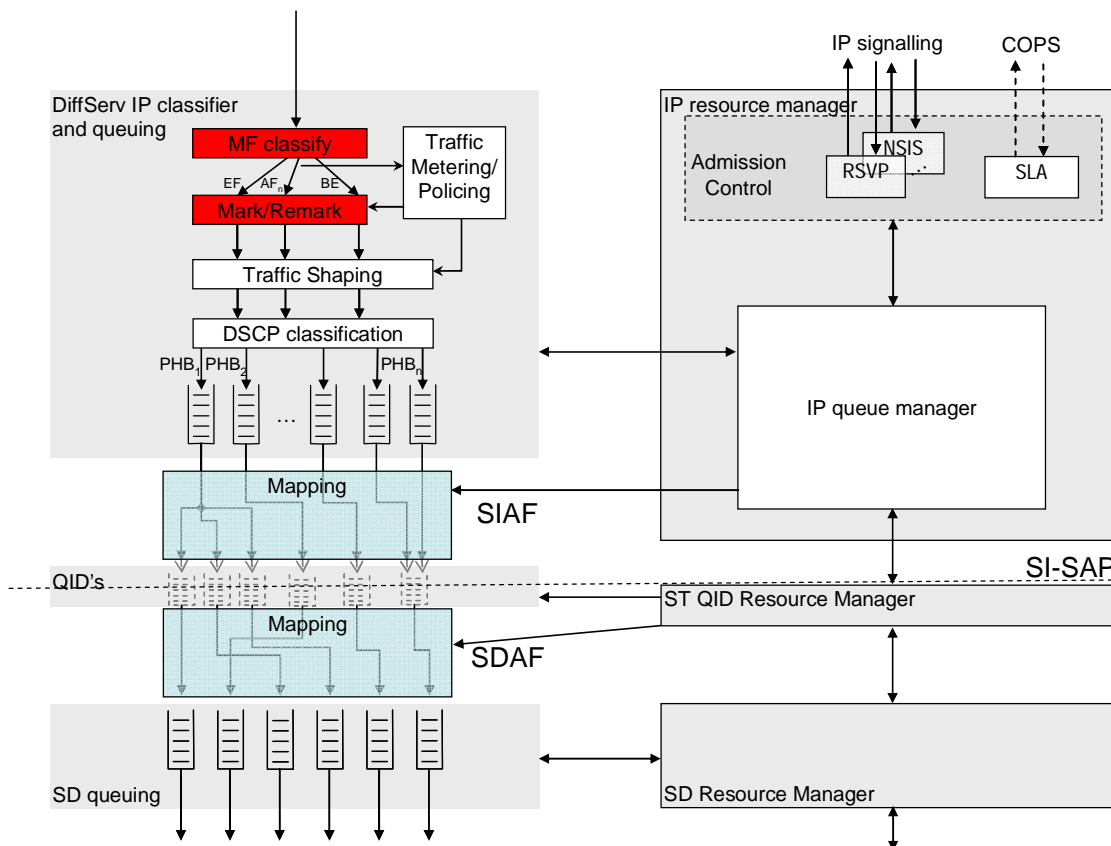
Figure 6: Control-Plane operations of a DiffServ-aware ST

## 5.2.2.1 Ingress ST DiffServ Functions

### 5.2.2.1.0 Detailed Functional Architecture

An Ingress DiffServ-aware ST is meant to be the ingress point of a DiffServ-aware BSM system, but not necessarily the ingress point of the BSM DS domain.

The functional architecture of the Ingress ST for DiffServ is as in figure 7.



**Figure 7: Detailed architecture of an ingress DiffServ-aware ST**

The Control Plane consists of one module above the SI-SAP (the IP resource manager), one module below (the SD resource manager), and one interfacing module (the STQRM), which performs the core functionalities and which is also functionally located below the SI-SAP. In particular the STQRM:

- 1) receives requests from the IP module to allocate, to release or to modify resources, through the set of primitives described in ETSI TS 102 463 [i.14];
- 2) translates these requests into QID allocation/release/modification actions;
- 3) checks with the SD module how the upper layer requests can be mapped to real resources.

In a DiffServ scenario the IP queues will most likely remain fixed to a limited number. Nevertheless the management of the queues might be dynamic. In particular an admission control module (and an interaction with a module responsible of SLAs), might dynamically check whether new IP flows are arriving at the ST. This might trigger the request for new resources, even if then on the user-plane the new IP flow will be aggregated with other existing ones in the same IP queue.

The request for new resource allocation might be also explicit, by means of IP signalling, such as RSVP or NSIS. The SLAs might be also dynamically changed, e.g. by means of COPS, SNMP, and other network management methods, and this can also trigger new requests from the IP resource manager to the STQRM. The interworking with QoS signalling protocols might then be envisaged in the IP resource manager, and should be handled by dedicated submodules, as shown in figure 6. The mentioned protocols (RSVP, NSIS) are taken as example and a brief discussion is given in the following two clauses.

The translation of the IP requests into QID operations (allocation/release/modification), and the mapping of the QIDs into SD queues, in the different (static and dynamic) cases, will be discussed in clause 6.

The operation of SD modules is system-dependent and will not be considered in the present document. The assumption is that allocation/release/modification of resources at SD layer will be viewed by the STQRM as allocation/release/modification of SD queues, and actions will be accordingly performed, e.g. remapping of QIDs to SD queues or allocation/release of QIDs.

If the SI layers issue a QID open/modify request which requires additional SD resources, an admission control procedure needs to be triggered at SD layers. The allocation of resources at SD layer may succeed or fail; a few more words need to be spent to explain what happens in these two cases.

If the resource allocation succeeds, one, or more, new QIDs can be allocated and DiffServ aggregated traffic can be sent over them. It should be noted that resources may be allocated by SD layers in an unsolicited manner, e.g. by means of management-plane signalling, but their association to IP services is done only if and when a request for QID allocation is sent by the SI layers. So it may happen, even if this will not be the common case, that, at the moment the QoS request is issued, the SD resources are already available; in this case a QID open/modify confirmation is sent back from the SD to the SI layer without the need of checking resources availability with the NCC.

If the resource allocation fails, two considerations should be made:

- A QoS request to provision a change of the SD resources does not modify the operation of DiffServ forwarding: Traffic may continue to be forwarded to the PHB which did not receive the requested modification; any excess traffic could be re-mapped to another acceptable PHB, or discarded, e.g. according to traffic conditioning agreements (TCAs).
- The failure to supply the requested resource should be indicated via the SI-SAP by means of the appropriate primitives (see ETSI TS 102 463 [i.14]). This may allow the SI layer to make an alternative QoS request that can satisfy the required SLA. When the request was made via a QoS provisioning protocol (e.g. RSVP, NSIS) the result of this allocation request can be returned to the originator. If insufficient resource can be allocated to meet the SLA, the BSM system should notify the management plane that there has been a provisioning failure.

#### 5.2.2.1.1 RSVP

RSVP is a signalling protocol that enable applications to request resources from the network, it is normally used in conjunction with the IntServ paradigm, so requested resources are reserved for single IP flows. Actually nothing prevents using RSVP together with DiffServ IETF RFC 2998 [i.4]. A network can react to RSVP requests by allocating resources based on aggregated flows. So it can respond by explicitly admitting or rejecting RSVP requests, exactly in the same way as an IntServ network can do. IP flows which are accepted will be then marked with a particular DSCP and they will be treated by the DiffServ domain as behaviour aggregates (BA).

This can be done in a BSM system, provided that the DiffServ-aware ST contains an RSVP agent, which is able to receive and respond to RSVP requests. The IP resource manager in the ST will have to perform the following operations:

- translate RSVP requests into QID allocation requests;
- forward the QID allocation requests to the STQRM;
- receive the reply from the STQRM and configure the IP queuing module in the user-plane (i.e. the traffic classification function) accordingly; e.g. if the flow was accepted the relevant packets can be marked with the relevant DSCP;
- send appropriate RSVP signalling.

The STQRM, which received a QID allocation request, performs the following operations:

- check for available SD resources to accommodate the request;
- if resources are available, make the required changes in the IP-to-QID or QID-to-SD mappings, and notify the IP resource manager;
- if SD resources are not available, request allocation of new resources to the SD resource manager (which will query the NCC), wait for response, make the required changes in the IP-to-QID or QID-to-SD mappings, and notify the IP queue manager.

#### 5.2.2.1.2 NSIS

The Next Steps in Signalling (NSIS) working group is considering protocols for signalling information about a data flow along its path in the network [i.5]. The NSIS suite of protocols is still in draft form, but it envisions support of various signalling applications that need to request or manipulate resource allocation in a network. The use of NSIS is also compliant with this BSM DiffServ architecture.

#### 5.2.2.2 Egress ST DiffServ Functions

An egress DiffServ-aware ST is meant to be the egress point of a DiffServ-aware BSM system, but not necessarily the egress point of the BSM DS domain.

On the satellite interface of the egress ST, no particular functions are required at SI-SAP and at SD level in a DiffServ-aware ST acting as egress node, either if the ST also acts as egress point of the BSM DS domain and if it does not. This means that QIDs and SD queues do not need to be implemented on the satellite receiving interface.

On the other hand on the terrestrial interface of the egress ST only traditional DiffServ operations should be implemented at IP level: The onward DiffServ behaviour to be applied to a packet is assumed to be fully defined by the IP header of the packet itself. If the egress ST also acts as egress point of the BSM DS domain, normal IP functionalities required by DiffServ egress nodes might be required: in particular DSCP re-marking or traffic conditioning functions, as defined by TCAs between the BSM DS domain and the peering DS domain which the ST connects to.

## 6 DiffServ QID Management

### 6.0 Aim and Scope

This clause defines how QID allocation shall be handled in case of DiffServ, and it describes the invocation, use and release of QIDs in resource reservation. These processes take place in the C-Plane of the SI-SAP.

QIDs are locally handled by the ST QID Resource Manager (STQRM). A centralized QID manager (BSMQRM) could be also adopted, it is the server for the whole BSM network, it is situated in the BSM central QoS Manager. It receives QID allocation requests from all STQRMs.

### 6.1 Description

QID management is the function that identifies the resource reservation to be done at lower layers, identifies the L2 queue(s), defines or modifies the properties of the abstract queue that is associated with that queue(s), assigns the Queue\_Identifier (QID) and makes the association with IP queue(s).

Once assigned, the QID is used for user data transfer via the SI-SAP, both for unicast and multicast IP flows; it is only required for sending data and is not required for receiving data. The actual data transfer on the U-plane does not involve QID management.

The satellite dependent layers are responsible for assigning satellite capacity to the L2 queues, and thus to the abstract queues, according to the specified L2 queue properties. This operations occur below the SI-SAP and they are hidden to the SI layers, so they also do not involve QID management.

The SI-SAP abstract queues, which can be used in DiffServ, are of two types:

- a) **Static queues:** These are queues that are created by management configuration and remain fixed for the whole duration of the ST working session. The parameters for these queues could be configured at ST logon by some means (e.g. sent by the NCC), or be statically stored in the terminal non-volatile memory, so that the ST will apply the same values at each logon session.
- b) **Dynamic queues:** These queues are created, modified and released dynamically using the primitives defined in ETSI TS 102 463 [i.14]. Queues can only be invoked by the satellite independent adaptation functions (SIAF) at the source ST by issuing the appropriate request primitive to the STQRM. The actual QID operation (creation, modification, release) is then performed by the STQRM. The allocation of such dynamic queues can be of hard or soft state, in the latter case it has to be refreshed to keep the QIDs alive.

Both types of QIDs can be locally (at the ST) or centrally (at the BSMQRM) managed.

Given the high number of users in a BSM network and the resulting high number of total active QIDs, it makes sense to delegate the management of some QIDs to the STs, and thus to have QIDs which are not centrally allocated. These QIDs might be reserved and released automatically by the ST (dynamic QIDs), in order to optimize capacity and resource utilization, e.g. on the basis of the actual traffic load, and present traffic classes (looking at the DSCPs), or they may be static.

## 6.2 QID Relationship to SD Resource Reservation

### 6.2.0 QID-to-SD Mapping

QIDs operate in close cooperation with the SD resource manager and they interface the SD queues to a set of IP queues. Normally, in case of DiffServ, the IP queues are limited to a small number (maximum number of allowed PHBs is 64, due to the 6 bits of the DS field), and they are usually static or slowly-changing (on a time scale of hours or days), since they handle aggregate flows.

The SD resource manager takes care of requesting and granting uplink transmission capacity; this operation might be static (the capacity for an ST is fixed or changing on a session basis) or highly dynamic (burst by burst requests, including e.g. BoD and DAMA functions, etc.). When the SD resources available at the ST change, the STQRM may also need to change their associations to QIDs, this might or might not have an impact on the QID-to-SD mapping (see ETSI TS 102 463 [i.14]). This operation is referred to as SD capacity control and this clause will address the two cases of:

- Static SD resources;
- Dynamic SD resources.

On the other hand the operations of reserving QIDs with defined properties, associating QIDs with DiffServ queues (the IP-to-QID mapping, as defined in ETSI TS 102 463 [i.14]), are not necessarily influenced by the SD operations of capacity control. Those operations, the actual QID management operations (as defined in the previous clause) occur across the SI-SAP, whereas SD capacity control occurs below it, thus the two can be considered independent from each other. For either static and dynamic SD resource management, both types of QIDs can be used:

- Static QIDs: they imply static DiffServ IP queues and static IP-to-QID mapping;
- Dynamic QIDs: they imply dynamic IP-to-QID mapping, but not necessarily dynamic DiffServ IP queue management, since, as already mentioned, DiffServ queue should be considered quasi-static.

Differently from the IntServ case, where only dynamic QIDs with dynamic SD resources are meaningful, in DiffServ all four cases make sense. They will be analysed in this clause 6.2. Anyway since SD capacity control operations, being located below the SI-SAP, are system and implementation dependent, the discussion of these four case will focus on the actual QID management operations (across the SI-SAP), in particular highlighting the impacts they have on the SI-SAP interface and on the definition of DiffServ queues. Impacts on the QID-to-SD mapping (below the SI-SAP) will be just briefly mentioned.

Of course the particular QID-to-SD mapping selected for one QID characterize its QIDSPEC. The list of QIDSPEC parameters to be used in case of QIDs interworking with DiffServ can be derived from ETSI TS 102 463 [i.14], the only difference being that DiffServ QIDs do not need to use the complete set of values described therein. The QIDSPEC values to be specified for each DiffServ class (BE, EF, AF, CS) are discussed and explained in clause 6.3. Example mappings of QIDs to SD resources for some BSM families are given in annexes A and B.

## 6.2.1 Static SD Resources

### 6.2.1.0 Overview

Static queues refer to the case where the uplink capacity assigned to each ST is fixed, or it is only changing on a session basis (it might be configured at logon upon communication from the NCC).

Even if the overall uplink capacity of each ST remains fixed, the ST may have multiple SD queues for different service classes. In this case the set of SD queues can be pre-configured in the ST by some means, and thus it is also static, or it can also be dynamically managed as far as the overall capacity assigned to the ST is not exceeded.

#### 6.2.1.1 Static QIDs

According to the SLA and a pre-configured set of DiffServ queues, a consistent IP-to-QID mapping is created and statically maintained. No signalling is required across the SI-SAP.

The QID-to-SD also has to be static. The way in which QIDs are (statically) associated to SD queues is left to the implementers and to the specifics of each SD technology.

It is suggested to have locally managed QIDs, but nothing prevents to have also centrally managed ones.

#### 6.2.1.2 Dynamic QIDs

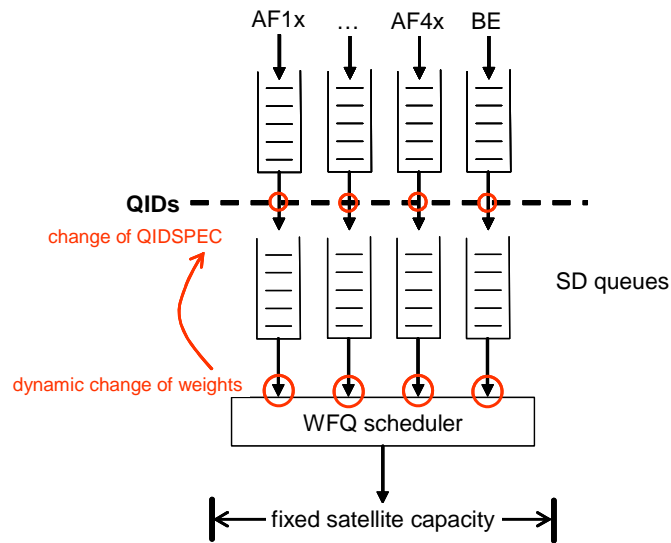
Either the IP-to-QID mapping and the QIDSPEC can be dynamically changed using the primitives defined in ETSI TS 102 463 [i.14].

Two cases can be identified for dynamic QID modification:

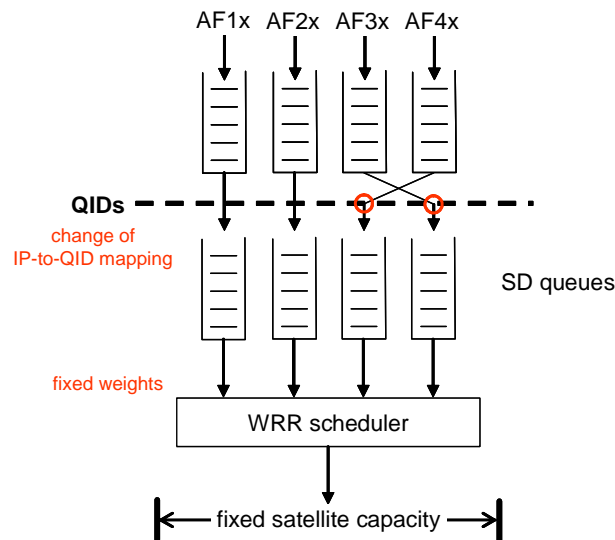
- 1) Modification request comes from SI layers (IP queue manager). The IP resource manager by means of explicit signalling (e.g. RSVP or COPS) or by monitoring the IP queue occupancy, may suggest modifications to the IP-to-QID mapping or to the QID QoS characteristics (QIDSPEC); this has to be done knowing that the overall transmission capacity remains fixed. This is communicated down to the STQRM, which, if accepted, adapts the QIDs to the new needs of the SI layers. The set of DiffServ queues remains normally unchanged.
- 2) Modification request comes from SD layers (STQRM). The QIDs can adapt to the varying incoming IP traffic: by means of feed-forward information coming from the SI layers (see SI-SAP primitive SI-C-QUEUE\_STATUS-res in ETSI TS 102 463 [i.14]) the STQRM tries to optimally exploit the fixed transmission capacity. In this case the STQRM should signal the QID modifications to the SI layers. Some remarks should be made:
  - a. IP-to-QID mapping should be dynamically changed only if this does not produce packet re-ordering within one IP flow;
  - b. Some modifications need not to be reflected in QID modifications, these types of modifications should not be signalled; for instance BE QID does not have an expected rate, so the rate R parameter in the QIDSPEC is normally undefined (see later in clause 6.3.1), if unused capacity of a rate-R QID is re-assigned to the BE effort class this should not change the R field in the BE QIDSPEC;
  - c. Some modifications might be temporary and highly dynamic; for example the unused capacity of one DiffServ PHB might be assigned to a lower-order PHB just for one time slot. It is suggested to avoid signalling these type of modifications if they occur on highly dynamic basis.

The QID-to-SD mapping can be either static or dynamic. It is static in the case the number of QIDs remains fixed, but other QID characteristics change, i.e. either the IP-to-QID mapping or the QIDSPEC. A typical way to modify the QIDSPEC can be dynamic scheduling, algorithms such as Weighted Fair Queueing (WFQ) that temporarily give other flows the unused bandwidth might be used (see figure 8); change of the IP-to-QID mapping might be used in case the SD scheduler has fixed weights (see figure 9). The QID-to-SD mapping is dynamic in the case the number of QIDs (and/or of SD queues) changes and so a QID-to-SD re-mapping is needed. It is recalled that QID-to-SD mapping does not need to be signalled across the SI-SAP.

Both types of QIDs, locally (at the ST) or centrally (at the BSMQRM) managed, can be used.



**Figure 8: QID-to-SD static mapping with dynamic change of the QIDSPEC**



**Figure 9: QID-to-SD static mapping with dynamic change of the IP-to-QID mapping**

## 6.2.2 Dynamic SD Resources

### 6.2.2.0 Overview

Dynamic queues refer to the case where the uplink capacity assigned to each ST is not fixed, and it might be changing on a highly dynamic basis (burst by burst requests, including e.g. BoD and DAMA functions, etc.).

The ST may also have multiple SD queues for different service classes. In this case the set of SD queues can also be dynamically managed. This can be done either locally, as far as the overall capacity assigned to the ST is not exceeded, or centrally, upon confirmation from the NCC.



### 6.2.2.1 Static QIDs

According to the SLA and a pre-configured fixed set of DiffServ queues, a consistent IP-to-QID mapping is created and statically maintained.

The set of QIDs is fixed, but the SD queues can dynamically change in number and characteristics (capacity, delay, etc.). So in this case the QID-to-SD mapping is dynamic.

The criteria with which SD resources are requested/release and the way in which QIDs are associated to SD queues are left to the implementers and to the specifics of each SD technology. It is recommended though to make use of the feed-forward information coming from the SI layers (see SI-SAP primitive SI-C-QUEUE\_STATUS-res in ETSI TS 102 463 [i.14]) on the status of the IP queues to make appropriate decisions on SD resources modifications. In any case if QIDs are static the QIDSPEC parameters cannot be changed; so the QID-to-SD re-mapping should be done in such a way not to produce modifications to the QIDSPEC parameter of a QID. An abstract queue can be re-mapped to a different SD queue only if the new QoS characteristics (capacity, transmission delay, etc.) are compliant with the ones expressed by the static QIDSPEC parameters.

Both types of QIDs, locally or centrally managed, can be used.

### 6.2.2.2 Dynamic QIDs

This is the most flexible scenario. As in the case with static SD resources, QIDs can be dynamically changed by modifying the IP-to-QID mapping and/or the QIDSPEC, using the primitives defined in ETSI TS 102 463 [i.14].

Again two cases can be identified for dynamic QID modification:

- 1) Modification request comes from SI layers (IP queue manager). The IP resource manager by means of explicit signalling (e.g. RSVP or COPS) or by monitoring the IP queue occupancy, may suggest modifications to the IP-to-QID mapping or to the QID QoS characteristics (QIDSPEC). Differently from the case with static SD resources this can be done knowing that the overall transmission capacity can be changed. The modifications is requested down to the STQRM, which, if accepted, adapts the QIDs to the new needs of the SI layers. The set of DiffServ queues remains normally unchanged.
- 2) Modification request comes from SD layers (STQRM). The QIDs can adapt to the varying incoming IP traffic: by means of feed-forward information coming from the SI layers (see SI-SAP primitive SI-C-QUEUE\_STATUS-res in ETSI TS 102 463 [i.14]) the STQRM tries to optimally exploit the assigned transmission capacity. Considering that the assigned transmission capacity can be changed, this scenario offers a very flexible set of possibilities for terminals and SD systems implementations. The STQRM should signal the QID modifications to the SI layers. Some remarks should be made:
  - a. IP-to-QID mapping should be dynamically changed only if this does not produce packet re-ordering within one IP flow;
  - b. Some modifications need not to be reflected in QID modifications, if they concern only SD resource management, these types of modifications should not be signalled;
  - c. Some modifications might be temporary and highly dynamic, it is suggested to avoid signalling these type of modifications if they occur on highly dynamic basis;
  - d. Some QID modifications can be deliberately hidden to the SI layers; e.g. capacity for some classes (e.g. BE) may be released if not needed, without explicit QID modification, the capacity might be requested again if new packets for that class arrive.

The QID-to-SD mapping can still be static or dynamic, as in the case with static SD resources. It is static in the case the number of QIDs remains fixed, but other QID characteristics change, i.e. either the IP-to-QID mapping or the QIDSPEC. A typical way to modify the QIDSPEC can be dynamic scheduling, but also increase of the transmission capacity. The QID-to-SD is dynamic in the case the number of QIDs (and/or of SD queues) changes and so a QID-to-SD re-mapping is needed.

Both types of QIDs, locally or centrally managed, can be used.

## 6.2.3 Summary

Table 3 summarizes for all combinations of static and dynamic QIDs and SD queues the corresponding possible ways of handling the two levels of mapping (IP-to-QID mapping and QID-to-SD mapping). It is recalled that dynamic QID-to-SD mapping does not need to be signalled across the SI-SAP, as it is implemented inside the terminal and it might be satellite dependent. The primitives defined in ETSI TS 102 463 [i.14] to create, modify and release QIDs need to be used only in case of dynamic QIDs.

**Table 3: QID configuration possibilities in case of static and dynamic QID and SD resources**

	Static SD resources		Dynamic SD resources	
	Static QIDs	Dynamic QIDs	Static QIDs	Dynamic QIDs
<b>IP-to-QID mapping</b>	static	static or dynamic	static	static or dynamic
<b>QID-to-SD mapping</b>	static	static or dynamic	static or dynamic	static or dynamic
<b>Use of SI-SAP C-plane primitives for QID setup</b>	No	Yes	No	Yes

## 6.3 QID Relationship to SI layers

### 6.3.0 DiffServ Per Hop Behaviours (PHBs)

The ST queuing model should support a typical DiffServ IP queue set for the following Per Hop Behaviours (PHBs):

- 1) Default (Best Effort - (BE), defined in IETF RFC 2474 [6]).
- 2) Assured Forwarding (AF, IETF RFC 2597 [8]).
- 3) Expedited Forwarding (EF, IETF RFC 3246 [9] and IETF RFC 3247 [10]).
- 4) Class Selector (CS, IETF RFC 2474 [6]).

DSCP values for each PHB should be assigned according to the IETF recommendations (IETF RFC 2474 [6], IETF RFC 3246 [9], IETF RFC 3247 [10] and IETF RFC 2597 [8]). Specific PHBs might be defined in the BSM DS domain, the associated DSCPs can be selected in the ranges provided for experimental or local use (EXP/LU see annex C) and should be compliant with the mentioned IETF guidelines.

Each PHB will normally be mapped to one separate IP queue, the PHB may be associated to one single DSCP or to multiple DSCPs (e.g. AF packets of the same class but with different drop precedences), which would be mapped onto the same IP queue.

Traffic Conditioning Agreement (TCA) rules, as part of the SLA, are enforced before packets are forwarded to lower layers. In particular:

- strategies for packet dropping should be defined in the TCA and applied if IP queues approach overflow conditions;
- actions to be applied to out-of-profile traffic should be defined (e.g. re-marking, discarding), in a way that packet re-ordering in one IP flow is avoided;
- dynamic re-mapping of IP queues to QIDs should be avoided within one IP flow in order to avoid packet re-ordering.

This guarantees that the SD layers can deal with SLA-conforming traffic only, and it should avoid queue overflow and packet drops at SD layers.

The IP queues handle aggregate IP flows by forwarding packets to the appropriate QIDs, according to the IP-to-QID mapping.

### 6.3.1 Best Effort

The IP flows which are mapped to this PHB have no expectations on the level of service received by the SD layers. So for BE traffic packets are directed to a BE FIFO queue, but without any conditioning. They remain in the queue until layer 2 resources are made available, as a result of layer 2 on capacity control and scheduling.

Nevertheless it is recommended to apply strategies to avoid starving the BE traffic queue. Two options can be identified:

- 1) A first option is to reserve some minimal SD resources for BE traffic, according to IETF RFC 2474 [6]: "A reasonable implementation of this PHB would be a queueing discipline that sends packets of this aggregate whenever the output link is not required to satisfy another PHB. A reasonable policy for constructing services would ensure that the aggregate was not starved. This could be enforced by a mechanism in each node that reserves some minimal resources (e.g. buffers, bandwidth) for Default behaviour aggregates".
- 2) Another possibility is to inform the SD layers on the load status of the BE queue in order to allow SD layers to optimize the use of unused resources or request additional bandwidth. An appropriate primitive is provided in ETSI TS 102 463 [i.14] for this purpose (the SI-C-QUEUE\_STATUS-res primitive), it is transmitted by the IP queue manager to the STQRM.

In general it is recommended to implement one BE IP queue and to map it to at least one BE QID (multiple BE QIDs might be implemented, e.g. to different destination BSM\_IDs). The BE QID(s) should be static, so that the IP-to-QID mapping for the BE PHB can be static, too, but this is not mandatory. The mapping of the single (or multiple) BE QID(s) to SD queues is left to system implementation and specific solutions. The QID-to-SD mapping can be static if SD resources and QIDs are static, but it can also be dynamic with three flavours:

- 1) Static SD resources with dynamic QIDs: the physical transmission capacity remains fixed, but the information on the variations in the IP queues occupancy (which are signalled from the SI layers down to the STQRM) may be used to optimize the use of SD resources: BE QID(s) might be temporarily re-mapped to SD queues where higher-order PHBs left unused capacity;
- 2) Dynamic SD resources with static QIDs: changes of the transmission capacity may trigger re-mapping of the BE QID(s) to different SD resources;
- 3) Dynamic SD resources with dynamic QIDs: changes of the transmission capacity may trigger re-mapping of the BE QID(s) to different SD resources, or release/opening of BE QID(s).

The QID associated to the BE class can optionally contain two parameters in its QIDSPEC: the rate R and the slack term S. The rate R represents the minimum provided transmission rate at the SD layer, measured in bytes of IP datagram per second. The slack term S defines the maximum expected packet queueing delay, i.e. the time the packet spends in the ST at the outgoing satellite interface waiting for being selected for transmission, this includes all components such as buffering delay, delay due to header processing, delay due to competition with other traffic classes, etc., and it may be very difficult to estimate for a BE QID. If the parameter R, or S, is specified the QID-to-SD mapping (either static or dynamic) should be done in such a way to guarantee the given R, or S, value.

When the BE IP queue is full, no particular algorithms need to be used to select the packets to be dropped: A drop tail mechanism may be implemented to drop the packets, or more sophisticated Active Queue Management (AQM) may be used to ensure better congestion behaviour in these conditions (e.g. the Random Early Detection algorithm [i.11]).

The QIDSPEC parameter of the BE QIDs can be empty.

The recommended DSCP for the default PHB with these specifications is the bit pattern '000000' [6].

## 6.3.2 Expedited Forwarding

The EF PHB is designed to provide low-loss, low-latency, low-jitter, assured bandwidth services, where packets normally encounter short or empty queues. Intuitively the service rate for EF traffic on the output satellite interface should be at least the configured rate  $R$ , independent of the offered load of non-EF traffic.

An EF-compliant ST should conform to the specifications given in IETF RFC 3246 [9] and IETF RFC 3247 [10], in particular for the formal definition expressed by the "packet-identity-aware" equations therein; the "aggregate-behaviour" equations are not considered relevant for an ST for two reasons:

- 1) an ST is not likely to have many interfaces;
- 2) one single model has to be chosen to define QID parameters, and the "packet-identity-aware" formal definition was selected for this purpose.

This formal definition [9] assumes that EF packets should ideally be served at rate  $R$  or faster, and bounds the deviation of the actual departure time of each packet from the "ideal" departure time of that packet, according to some equations.  $E_p$  is defined as the error term for the treatment of individual EF packets, it represents the worst case deviation between the actual departure time of an EF packet and the ideal departure time of the same packet.

Following this "packet-identity-aware" definition, an EF-compliant ST should be able to characterize the range of possible  $R$  values that it can support on its satellite interface while conforming to the equations in IETF RFC 3246 [9], and the value of  $E_p$  that can be met on the interface for each value of  $R$ .

So an ST should be able to estimate  $E_p$  for every EF flow. A variety of factors in the implementation of an ST will influence the value of  $E_p$  (e.g. output schedulers and internal node design). These factors are discussed in IETF RFC 3246 [9] and IETF RFC 3247 [10], where also values for a variety of widely used class-based schedulers are given. An  $E_p$  value of "undefined" (N/A) may be also specified in some cases.

Given a known value of  $E_p$  and a knowledge of the characteristics of the EF traffic offered to the satellite output interface, it is possible to bound the delay and jitter that will be experienced by the EF traffic. The delay bound is given by:

$$D = b/R + E_p ;$$

where:

- $R$  is the configured EF service rate on the satellite interface;
- the total offered load of EF traffic destined to the satellite interface, summed over all input interfaces, is bounded by a token bucket of rate  $r \leq R$  and depth  $b$ .

$D$  also provides a bound on the delay jitter.

These parameters ( $R$ ,  $E_p$ , and optionally  $r$  and  $b$ ) should be used in the EF QID definition respectively for the values of the following fields in the QIDSPEC parameter:

- Rate [ $R$ ] (32-bit IEEE floating point number): it is expressed in bytes of IP datagrams per second (including IP headers), and can range from 1 byte per second to as large as 40 terabytes per second;
- Slack Term [ $S$ ] (32-bit integer): it is expressed in microseconds, the value "undefined" (N/A) may be specified for very particular cases (e.g. non-FIFO hierarchical scheduling, see IETF RFC 3246 [9]);
- Token Bucket Rate [ $r$ ] (32-bit IEEE floating point number): same format as  $R$ , it is measured in bytes of IP datagrams per second;
- Token Bucket Size [ $b$ ] (32-bit IEEE floating point number): it is also measured in bytes and can range from 1 byte to 250 gigabytes.

The  $R$  and  $S$  parameters associated to that QID represent the real characteristics of the layer-2 resources allocated to that QID, the  $r$  and  $b$  parameters represent the properties of the EF traffic allowed for that QID (the associated EF traffic has to be bounded by a token bucket of rate  $r \leq R$  and depth  $b$ ). So the EF IP queue should be associated to an appropriate QID with the characteristics just mentioned, and consequently the QID-to-SD mapping should be done in such a way to guarantee the required QIDSPEC values.

It is possible to define multiple instances of EF in one ST. Each instance should be associated to a separate IP queue and to an associated QID with the appropriate R and S values. Having multiple instances of EF has some effects on the  $E_p$  value of each instance, these effects will depend considerably on how the multiple instances are implemented (e.g. type of scheduler, etc.).

The ST implementation of EF should include some means to limit the damage EF traffic could inflict on other traffic (e.g. by avoiding unlimited pre-emption of other traffic).

The recommended DSCP for the default PHB with these specifications is the bit pattern '101110' [9].

### 6.3.3 Assured Forwarding

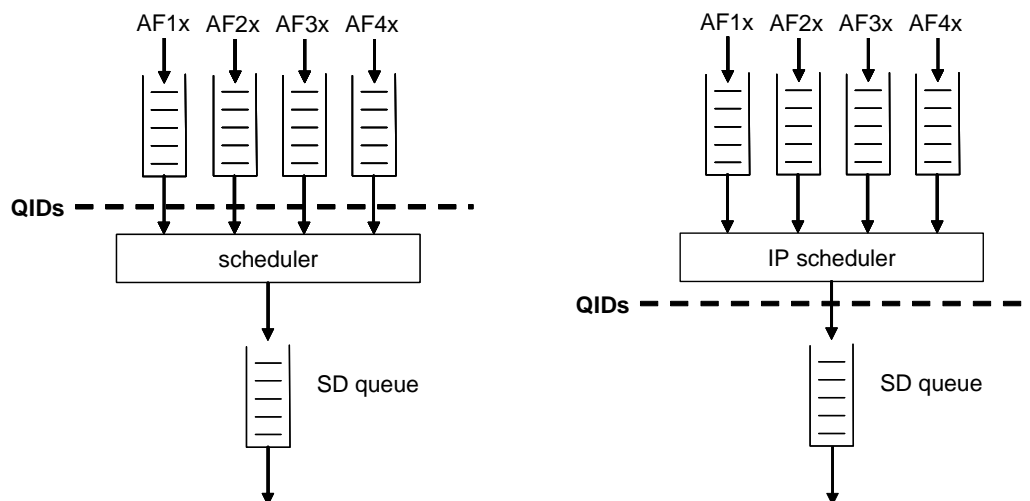
IETF defines four independently forwarded AF classes, within each class one of three different levels of drop precedence can be specified.  $AF_{ij}$ , with  $1 \leq i \leq 4$  and  $1 \leq j \leq 3$ , represents the DSCP for AF class  $i$  with drop precedence  $j$ . It is recommended to support at least one AF class with two drop precedence levels.

It should be understood that AF is a type of PHB group, and each AF class is an instance of the AF type [11]. Each AF class in each ST should be allocated a separated IP queue. Each AF class should be assigned a certain amount of forwarding resources (buffer space and bandwidth) at SD layers, by defining an absolute bandwidth or a percentage of the interface bandwidth. This can be done in two ways:

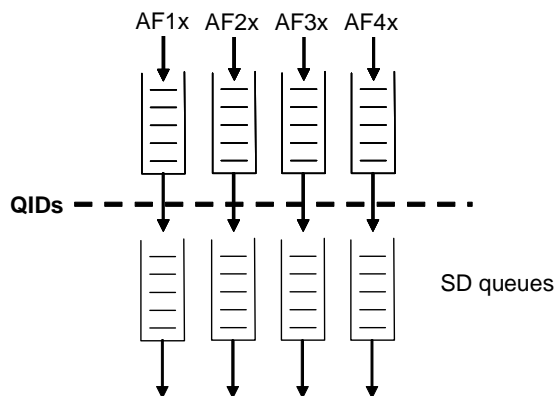
- 1) a scheduler below the AF queues can allow the bandwidth to be shared among the various classes defined, the output of the scheduler is then mapped onto an SD queue with given properties (bandwidth, transmission delay, etc.);
- 2) each AF queue can be mapped onto a separate SD queue with given properties, a scheduler below the SD queues may be optionally foreseen to access the physical layer resources.

The former case is represented in figure 10; scheduling algorithms like Class Based Weighted Fair Queuing (CBWFQ) allow the bandwidth to be shared among the various classes defined. The QIDs can be considered either before the scheduler or below, this only makes a difference in the way QIDs are managed.

The latter case is shown in figure 11, in this case QIDs represent in their QIDSPEC parameters the real properties of the SD queues.



**Figure 10: AF classes architecture with schedulers**



**Figure 11: AF classes architecture without scheduler**

AF classes should also be configurable to receive more forwarding resources than the minimum when excess resources are available either from other AF classes or from other PHB groups. This can be done with dynamic QID and/or dynamic SD resource management (see clause 6.2) by making use of the feed-forward information coming from the SI layers (see SI-SAP primitive SI-C-QUEUE\_STATUS-res in ETSI TS 102 463 [i.14]) on the status of the AF queues.

In any case the QID associated to one particular AF class should be defined by a rate  $R$  parameter in the QIDSPEC, which represents the minimum provided transmission rate at the SD layer, and the QID-to-SD mapping (either static or dynamic) should be done in such a way to guarantee the given  $R$  value. The specification of the slack term  $S$  in the QIDSPEC of an AF QID is optional. In case  $S$  is specified, it defines the maximum expected packet queuing delay, i.e. the time the packet spends in the ST at the outgoing satellite interface waiting for being selected for transmission (same definition as in the BE case, see clause 6.3.1). The  $S$  term may be difficult to estimate, for this reason is considered optional.

Within each AF class IP packets are marked with one of three possible drop precedence values. In case of congestion, an ST should try to protect packets with a lower drop precedence value from being lost by preferably discarding packets with a higher drop precedence value. In this respect AQM mechanisms for queuing and packet discarding at IP level should be supported on each AF queue, according to the guidelines given in IETF RFC 2597 [8] clause 4, e.g. by adopting Random Early Detection (RED) or Weighted RED (WRED). IP packets of the same AF class, but with different drop precedences, are mapped into the same IP queue, so they cannot be mapped onto different QIDs. Actually this would not make sense as packet discarding is applied at IP level, and there is no way to differentiate QIDs on the basis of IP packet dropping probability in the QIDSPEC (see also ETSI TS 102 463 [i.14], clause 6.2.1).

Packets belonging to the same AF class should not be re-ordered regardless of their drop precedence.

Recommended DSCPs for the AF classes are given in annex C.

### 6.3.4 Class Selector

These CS PHBs ensure that DS-compliant nodes can co-exist with IP-Precedence aware nodes. These CS codepoints can co-exist with the AF and EF ones, and thus they can be used in a BSM DS domain. They can be used to define additional PHBs specific of a BSM DS domain. The PHBs mapped to by these codepoints may have a detailed list of specifications, but they shall meet the following minimum requirements IETF RFC 2474 [6]:

- PHBs selected by a CS DSCP should give packets a probability of timely forwarding that is not lower than that given to packets marked with a CS DSCP of lower order (i.e. of smaller numerical value), under reasonable operating conditions and traffic loads;
- PHBs selected by distinct CS DSCPs should be independently forwarded, that is, packets marked with different CS DSCP may be re-ordered.

CS PHBs are not specified, so the associated IP queues might be of any type, static or dynamic, and for this reason the associated QIDs might be static or dynamic, too. In this respect CS PHBs can be managed both with static or dynamic SD resources, too, according to system specifics.

For a QID associated to one particular CS it is suggested to specify the rate R parameter in the QIDSPEC; it should represent the minimum transmission rate at the SD layer provided to that class. The QID-to-SD mapping (either static or dynamic) should be done in such a way to guarantee the given R value. The specification of the slack term S in the QIDSPEC of a QID for this class is optional.

The codepoints DSCP = 'xxx000' represent the set of CS DSCPs, the codepoint '000000' shall be preserved for the default PHB (BE), the codepoints '11x000' may be retained for network control traffic (e.g. routing).

### 6.3.5 Summary

Table 4 summarizes the QIDSPEC parameters to be used for each DiffServ PHB.

In general DiffServ QIDs (except for the best effort QIDs, see clause 6.3.1) should at minimum be described by the value "rate [R]" in the QIDSPEC, the remaining values are optional. R represents the QID minimum admitted rate measured in bytes of IP datagrams per second, R can represent the link transmission rate or a share of it. The BE QIDSPEC is the only one which can be empty (see clause 6.3.1).

The slack term S parameter should be always specified in case of EF QIDs; it can assume the value "undefined" (N/A) in very particular cases (e.g. non-FIFO hierarchical scheduling, see IETF RFC 3246 [9]).

**Table 4: QIDSPEC parameters to be used for each PHB**

QIDSPEC parameters	Per Hop Behaviours			
	BE	EF	AF	CS
<b>Token Bucket Rate</b> [r]		O		
<b>Token Bucket Size</b> [b]		O		
<b>Peak Data Rate</b> [p]				
<b>Minimum Policed Unit</b> [m]				
<b>Maximum Packet Size</b> [M]				
<b>Rate</b> [R]	O	M	M	M
<b>Slack Term</b> [S]	O	M	O	O
NOTE: M: mandatory parameter O: optional parameter empty cell: parameter not required				

## Annex A (informative): Relationship to the RSM-B Air Interface family (DVB-RCS)

### A.0 General

The present annex contains a reference example for DiffServ implementation over a BSM family (the Regenerative Satellite Mesh - B, RSM-B, ETSI TS 102 429-1 [i.9], which is compliant with the described specifications. The example is given for the purpose of clarification and it should be considered as possible embodiment; so it should not limit the applicability of the system to other solutions.

This annex addresses the BSM Bearer Service as a component of the complete end-to-end QoS architecture in the context of SATLIFE DVB-RCS regenerative system [i.6]. In this annex the SATLIFE QoS model is described, showing how QIDs can be used in this architecture.

### A.1 SATLIFE satellite system summary

SATLIFE is based on the AmerHis system which is basically a regenerative DVB payload on board of the Amazonas satellite [i.7]; both SATLIFE and AmerHis follow the system definition given by RSM-B BSM family. This system integrates and combines both DVB-S and DVB-RCS into one unique regenerative and multi-spot satellite system. The DVB-RCS return channel standard is employed by all users to access the satellite through a standard uplink. On board, the regenerative payload multiplexes the traffic from diverse sources into one or more DVB-S data streams capable of being received by any standard DVB-S receiver. The on-board repeater is not only capable of multiplexing signals from the same uplink, but also cross-connecting and/or broadcasting channels coming from separate uplink coverage areas to different downlink coverage areas. The SATLIFE/AmerHis system is completed by a fleet of standard DVB-RCS terminals (RCSTs), a MS (Management Station) acting as NCC, and several RSGWs (Regenerative Satellite GateWays) for interconnection with terrestrial networks. AmerHis supports star (RCST to RSGW) and mesh (RCST to RCST) connectivity in just one satellite hop with unidirectional or bi-directional, point-to-point and point to multipoint connections. The connection control, QoS assurance, addressing resolution and support of multicast require the usage of a signalling protocol between the satellite terminals and the NCC: The Connection Control Protocol (C2P) [i.12] aims at assessing and enhancing the control plane of AmerHis DVB-RCS system. The NCC performs management functions and bandwidth allocation for the integrated system. This regenerative scheme is shown in figure A.1.

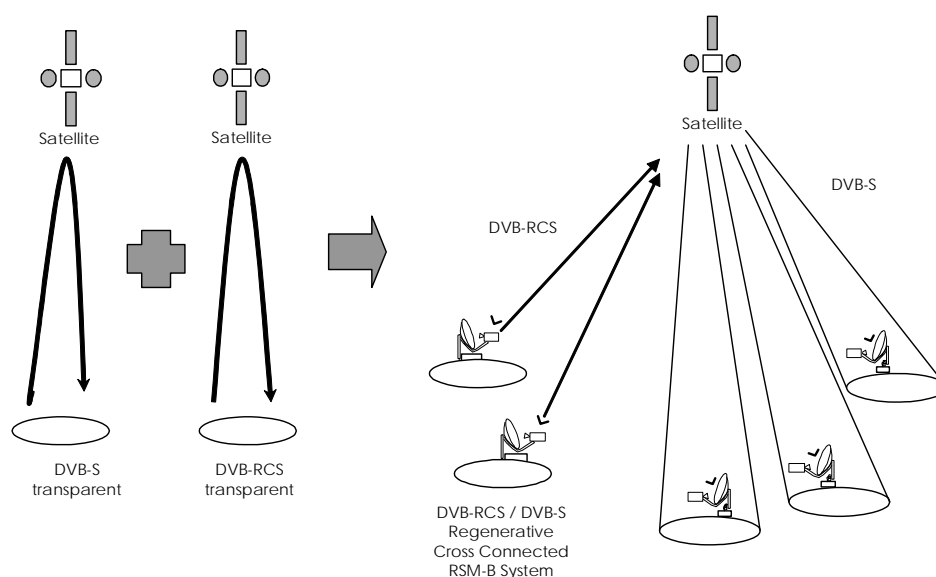


Figure A.1: RSM-B regenerative concept



## A.2 SATLIFE QoS model

Different types of IP services are supported in the SATLIFE/RSM-B system; from the most QoS sensitive services (conversational and interactive), to the best effort services. The SATLIFE QoS model follows the DiffServ approach as defined in ETSI TS 102 462 [2]. Layer-4 and layer-3 parameters are used to estimate layer-2 QoS parameter. Table A.1 shows an example.

The network layer QoS mechanisms performs prioritization of IP flows by asking the satellite link control layer for the most suitable transmission parameters for the considered application. QoS on RCST level is based on the adaptation of the satellite connection parameters to what the application requires. This requires identifying each application type and managing each of the application flows. The QoS mechanisms on the RCST are based on traffic differentiation. The RCST is capable of classifying IP traffic into several IP flows. Each IP flow is identified thanks to a MF filter definition. The queuing and scheduling between IP flows depends on the QoS strategy defined within the RCST.

**Table A.1: SATLIFE QoS parameters**

Layer-4 and layer-3 parameters	Layer-2 QoS connection parameters
Source IP address range	Activity timer
Destination IP address range	Priority (LP/HP/HPj)
DSCP range	Tx Sustained Data Rate
Protocol	Tx Peak Data Rate
Source TCP/UDP port	Rx Sustained Data Rate
Destination TCP/UDP port	Rx Peak Data Rate
	Direction (Unidir/Bidir)

An RCST is statically configured with concrete values to distinguish the different flows: source IP address range, destination IP address range, DSCP range, protocol, source TCP/UDP port, destination TCP/UDP port. Depending on the type of flow it will select the following parameters for the communication: priority (LP-Low Priority / HP-High Priority / HPj-High Priority jitter sensitive), transmitter sustained data rate, transmitter peak data rate, receiver sustained data rate, receiver peak data rate and direction (unidir/bidir).

SATLIFE terminals will perform a mapping between layer-3 parameters of IP traffic and link layer connection parameters. This mapping is manageable from the MS, being possible to configure up to 20 entries per RCST. Flow classification allows to have, simultaneously, different QoS levels for every service the subscriber is using.

Connections at data link layer level are established using the C2P protocol, proposed in [i.8], and being standardized under the BSM RSM-B family [i.12]. The basic characteristics of C2P are the following:

- 1) It is used to establish connections for transmitting/receiving IP traffic.
- 2) According to the originator of the connection request message, connections can be either on-demand (RCST-initiated) or permanent (NCC-initiated).
- 3) Connection request message sent by calling terminal includes information about IP addressing, bandwidth profiles, direction (uni or bi-directional), priority and type (uni or multicast): It is possible to map IP info of the incoming packets to certain connection types.
- 4) NCC performs connection admission control and maps traffic to channels, but terminals may route traffic on existing connections, avoiding unnecessary signalling traffic towards the NCC.
- 5) An RCST may have more than one connection (each one with different priority) with the same destination RCST.
- 6) C2P supports embedded address resolution.
- 7) C2P allows multicast group management.

The main differences of the SATLIFE QoS architecture with respect to the one defined in ETSI TS 102 462 [2] are the following:

- 1) Address resolution, routing information, and connection handling with other RCSTs is based on the RSM-B C2P protocol [i.12].
- 2) Several DVB-RCS Channel-id's are used: The Channel-id is used on the uplink link, typically to identify resources associated with a destination beam or multicast group. The channels define partitions of the uplink link resources that are treated independently from resources in other channels. So, an RCST can transmit data flows over different channels which are managed independently, either for connectivity or for QoS purposes.
- 3) MPE/MPEG2-TS encapsulation is used. MPE section packing is used so that the payload portion of the MPEG packet is fully used, leading to a better satellite bandwidth utilization.
- 4) PIDs are used for addressing purposes, each connection is identified with a different pair of PIDs. This allows multiplexing of different connections over the same Channel-id.

The points 1 and 2 should be carefully considered in the definition of the SI-SAP interface.

C2P is a connection oriented approach and it needs to be mapped to the SI-SAP interface. On the other hand, at the SI-SAP, QIDs may also have a soft-state nature, which means that they expire if not refreshed. This point needs to be carefully considered. Each soft-state QID which is associated to a C2P connection needs a regular refresh operation until the C2P connection is alive.

The usage of several Channel-id's at the RCSTs might be seen at IP level as different network interfaces where IP flows/classes are queued and scheduled independently; also the resources of each channel have to be managed independently. A possible approach is to identify the channel using the BSM\_ID, and to deal with independent IP flows in different QIDs.

Figure A.2 shows the relationship between IP flows, C2P connections, PIDs and Channel-ids.

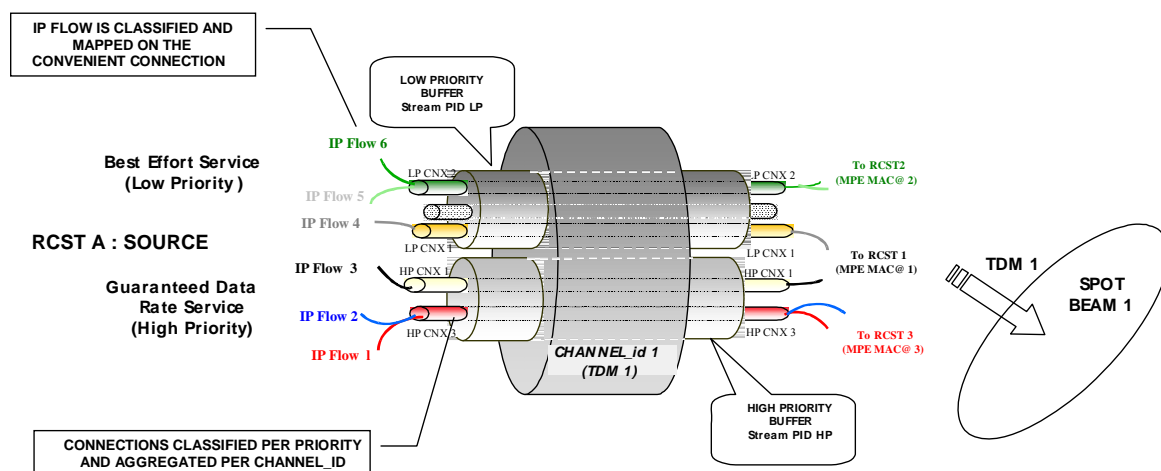


Figure A.2: Connection, channel\_ID and PIDs arrangement

## A.3 RCST/NCC Model

### A.3.0 Overview

The model of the ST behaviour is based on the BSM architecture, the SD includes MAC fragmentation and scheduling of different MAC queues (PIDs). The model represented in figure A.3 only shows one channel-id, because the resources are managed independently on each channel-id. The thick blue dotted line shows where the SI-SAP can be located, and so where the QIDs should be considered, for user and control planes. This approach represents a case of *dynamic SD resources* with *static QIDs*, according to the definition given in clause 6.2.2.1. In this case a possible mapping of QIDs to MPEG PIDs will be a one-to-one mapping: Each QID is mapped to a single PID.

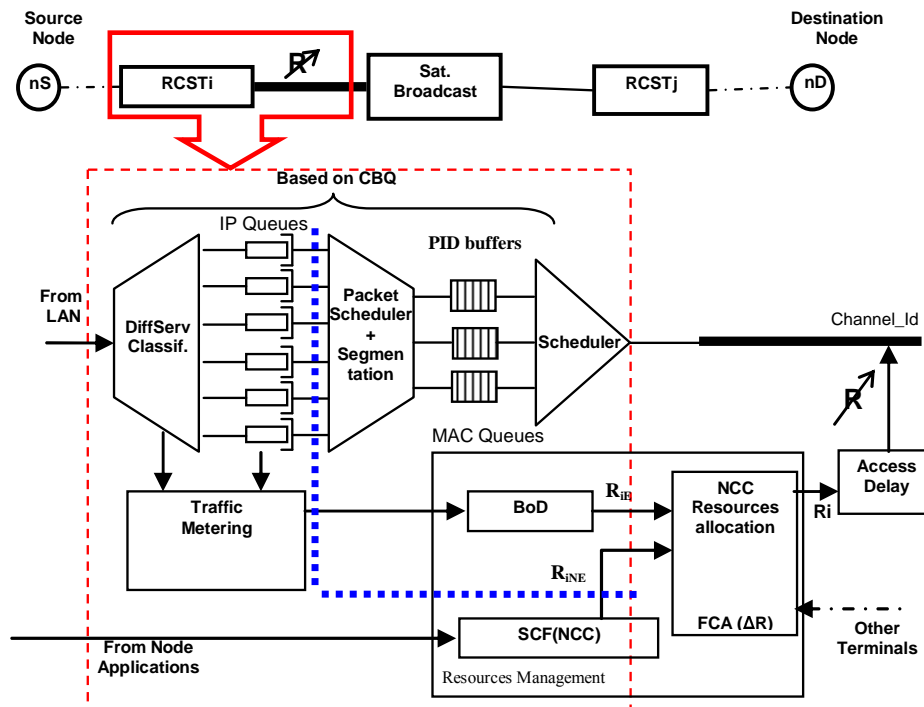


Figure A.3: SATLIFE/AmerHis RCST/NCC model

The following symbols are used in figure A.3:

$R_{iE}$ : RCST<sub>i</sub> bandwidth requirements for elastic traffic (TCP based).

$R_{iNE}$ : RCST<sub>i</sub> bandwidth requirements for EF non-elastic traffic.

$R_i$ : Bandwidth allocated to the RCST<sub>i</sub> by the NCC.

In the case of no congestion  $R_i = R_{iNE} + R_{iE} + FCA$ . In the case of congestion  $R_i = R_{iNE} + (R_{iE} - \Delta R)$ .

$\mathcal{R}$ : Bandwidth available at the terminal (this is variable, equal to  $R_i$ , but after the access delay).

### A.3.1 User Plane

The elementary building blocks enabling DiffServ include classifiers, traffic conditioners and scheduling. Figure A.3 shows the details of this architecture and the operations that are made when packets are transmitted to a shared channel (Channel-id). The first operation is the classification of IP packets from all the possible source nodes attached to the ST, based on the configured QoS policy. Packets are enqueued while maintaining statistics about them (traffic metering).

Layer 3 prioritization and classification are based on 4 IP queues. An example for traffic classification could be as follows:

- VoIP flows: classified as EF (HP<sub>j</sub>);
- Videoconferencing flows: classified as EF (HP<sub>j</sub>);
- HTTP flows: classified as AF (HP<sub>j</sub>);

- remaining flows: classified as BE (LP).

After classification, the packets are directed to FIFO IP queues with a configurable length and drop strategy.

They remain in the queue until level 2 resources are made available. Packets are scheduled according to the IP queue priorities defined in QoS policy as well (EF, AF and BE) or C2P connection priorities (HP<sub>j</sub>, HP, LP).

Given this limited bandwidth environment, scheduling at IP packet level results in an unacceptable delay/jitter for high priority real-time traffic. Performing another scheduling decision after layer-2 fragmentation will provide a significant QoS improvement. In order to do this multiple PIDs have to be used on the same channel-id, since MPEG frames of different IP packets cannot be multiplexed over the same channel and PID. So two different approaches can be adopted:

- use different channels for distinct delay-sensitive flows, e.g. VoIP (each channel-id has its own resources which are independent from other channel-id's);
- share the same channel among heterogeneous flows and configure properly the priorities.

With the former approach the link utilization is quite poor. On the other hand with the latter link sharing approach, the link utilization improves significantly, but separate PIDs (MAC queues) have to be used for VoIP traffic, otherwise the jitter for this delay-sensitive traffic becomes very high. The mapping between IP queues and the MAC PID queues and priorities should be kept flexible in order to support both approaches. In this respect the PID can be considered as SD resources which can be mapped onto the IP queues by means of QIDs. So the possibility to have QIDs with dynamic mapping to IP and SD queues enables the support of both approaches. The system is flexible, as it will let the network operator the final the decision on the channel utilization. A basic approach is to avoid HP<sub>j</sub> traffic to be mixed with other priority traffic. Only LP and HP traffic can be mixed together.

## A.3.2 Control Plane

Concerning the control plane, important issue is constituted by the capacity assignment strategies; in this context the SATLIFE example can be taken as a general case. The Bandwidth-on-Demand (BoD) algorithms and the Free Capacity Assignment (FCA) policy regarding spare capacity distribution are two mechanisms that impact the system performance at different traffic conditions: BoD is effective at high levels of traffic load (congestion) whereas FCA has an impact at low traffic load.

BoD is a crucial research topic, it has been investigated with many approaches, and several algorithms exist. FCA means that when free time slots are left after the demanded allocation, they are redistributed fairly among the terminals. Different BoD and FCA algorithms may have different goals and their performance might be evaluated according to different parameters: Fairness (fair distribution of resources between the terminals), Stability/Quick convergence, Efficiency ("optimum" resource utilization), etc.

Under no congestion, the analysis should be focused on the algorithms' stability/quick convergence to the limits imposed by SLA or TCP protocol over Satellite, but in general there is no optimum solution, which should be standardized. So every implementer should be left free to adopt the preferred one. As an example, for AF/BE elastic (TCP-based) traffic, the BoD algorithm could predict the bandwidth requirements based on the current status of the queues and their dynamics (traffic metering); bandwidth requirements for EF non-elastic traffic can be defined at session level based on explicit signalling. NCC resource controller should be aware of these two types of bandwidth requests in order guarantee EF resources (after admission control) even in a congested network.

In conclusion an RCST compliant with the BSM QID model provided in the present document is also enabled to use different algorithms for the capacity assignment on each channel-id (FCA, VBDC, RBDC, etc.): QIDs can be associated to different types of capacity assignment. This is important, since this issue is crucial in satellite systems; it addresses the problem of providing the required QoS while guaranteeing a high exploitation of the valuable satellite bandwidth, in contrast to wired network, where capacity over-provisioning is often used to ensure QoS for packet based transport. So the present document leaves the freedom to choose the preferred solution in this respect.

---

## Annex B (informative): Relationship to the TSS-A Air Interface Family (DVB-RCS)

The present annex contains a reference example for DiffServ implementation over a BSM family (the Transparent Satellite Star - A, Sub-family 1, TSS-A1, [i.13]), which is compliant with the described specifications. The example is given for the purpose of clarification and it should be considered as possible embodiment; so it should not limit the applicability of the system to other solutions.

This annex refers to "SatLabs System Recommendations v2.0" [i.10]. The Request Class (RC) concept introduced in that document is fundamental for the application of the QID concept.

An RC is a representation of a PHB at the MAC layer. It defines a behaviour of the MAC layer for a given aggregation of traffic. This behaviour includes a combination of capacity categories associated to the RC and a priority with respect to the other RCs supported by a DVB-RCS terminal (RCST). As an example, an RCST can have two RCs using Volume Based Dynamic Capacity (VBDC) only: one has a priority over the other. An RC is a concept similar to the PHB, but seen from layer 2. An RC can be seen as part of a PHB instantiation, since it will strongly influence the traffic forwarding.

RCs are associated to capacity requests in each channel-id, and they represent the embodiment of the SD resources and SD queues described in the present document. The association is centrally managed by the NCC, but the RCSTs are free to assign RCs to VPI/VCI or PIDs within one channel, and to re-assign exceeding capacity in one channel to different RCs associated to other channels; so the association can be locally changed if this has no impact on the central management. This way of operating RCs is like having dynamic SD resources (according to the terminology used in the present document), so with a dynamic QID-to-SD mapping which can be locally managed.

On the other hand the association between RCs and PHBs in the SatLabs proposal is static at the moment, even if the present document states that additional mappings may be configured by other means. It also said that the NCC should explicitly instruct the RCST of which RC should be used for each PHB, by assigning a corresponding RC to each of these PHBs. The RCST management parameters allowing such configuration are all static at the moment, which means that the RC have fixed characteristics. So it is possible to define *static QIDs*, with *static IP-to-QID mapping*, and with dynamic SD resource management (*dynamic QID-to-SD mapping*). As it was explained in the present document, the association between QIDs and SD resources remains below the SI-SAP and so it does not have to be signalled across the SI-SAP, so this way of operating RCs is compliant with the present TS and it does not imply the use of the SI-SAP primitives for the QID setup.

The SatLabs group may envisage, in the future, to define mechanisms for setting some RC configuration parameters dynamically. Both cases, dynamic and static RCs association to PHBs, are supported by the QID management presented in the present document (*dynamic QIDs*, with *dynamic IP-to-QID mapping*). Primitives exist to signal the QIDs (and thus RCs) characteristics to the IP layers, these primitives should be thus taken into account in the future when handling dynamic RCs.

RC identifier is proposed to be 4 bits, using the 4 bits defined for the Channel-id, whereas QID is 24 bits. In addition RC identifiers are sent over the air interface and they are involved in a system-specific signalling with the NCC, whereas QIDs are not expected to be sent over the air interface and they are only meant to be used locally in the ST. This means that a mapping between RCs and QIDs should be performed locally in the RCST once the resources have been allocated. This mapping is left to the implementers and to the terminal specifics, but in any case the high number of bits available for the QIDs and the possibility to use the QID management primitives gives freedom for real implementations and bigger potentialities to the use of RCs.

## Annex C (informative): DiffServ DSCPs

The present annex recaps the DiffServ codepoint recommended by IETF, which should be also adopted in BSM systems.

**Table C.1: Summary of DSCPs**

	PHBs	DSCPs			No.
<b>IANA assigned</b>	BE	000	00	0	1
	CS	001-111	00	0	7
	AF	001-100	01-11	0	12
	EF	101	11	0	1
<b>Not assigned</b>		000	01-11	0	3
		101	01-10	0	2
		110-111	01-11	0	6
<b>EXP/LU</b>		xxx	x1	1	16
<b>EXP/LU</b>		xxx	x0	1	16
<b>Total:</b>					64
<b>NOTE:</b>	Initially available for experimental or local use, but may be used for future standards action allocations as necessary.				

**Table C.2: Summary of AF DSCPs**

	AF1x	AF2x	AF3x	AF4x
<b>Low drop precedence (AFx1)</b>	001010	010010	011010	100010
<b>Medium drop precedence (AFx2)</b>	001100	010100	011100	100100
<b>High drop precedence (AFx3)</b>	001110	010110	011110	100110

---

## Annex D (informative): Bibliography

IETF RFC 4774: "Specifying Alternate Semantics for the Explicit Congestion Notification (ECN) Field" S. Floyd, November 2006.

---

## History

<b>Document history</b>		
V1.1.1	January 2007	Publication
V1.2.1	December 2015	Publication