ETSI TS 102 771 V1.1.1 (2009-06)

Technical Specification

**Digital Video Broadcasting (DVB);
Generic Stream Encapsulation (GSE)
implementation guidelines**

European Broadcasting Union   Union Européenne de Radio-Télévision

EBU·UER

DVB
Digital Video
Broadcasting

ETSI

Reference
DTS/JTC-DVB-243

Keywords
digital, DVB, TV

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

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or
perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF).
In case of dispute, the reference shall be the printing on ETSI printers of the 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:
http://portal.etsi.org/chaircor/ETSI_support.asp

Copyright Notification

ETSI

Contents

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://webapp.etsi.org/IPR/home.asp).

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 Joint Technical Committee (JTC) Broadcast of the European Broadcasting Union (EBU), Comité Européen de Normalisation ELECtrotechnique (CENELEC) and the European Telecommunications Standards Institute (ETSI).

NOTE: The EBU/ETSI JTC Broadcast was established in 1990 to co ordinate the drafting of standards in the specific field of broadcasting and related fields. Since 1995 the JTC Broadcast became a tripartite body by including in the Memorandum of Understanding also CENELEC, which is responsible for the standardization of radio and television receivers. The EBU is a professional association of broadcasting organizations whose work includes the co ordination of its members' activities in the technical, legal, programme making and programme exchange domains. The EBU has active members in about 60 countries in the European broadcasting area; its headquarters is in Geneva.

European Broadcasting Union
CH 1218 GRAND SACONNEX (Geneva)
Switzerland
Tel: +41 22 717 21 11
Fax: +41 22 717 24 81

Founded in September 1993, the DVB Project is a market led consortium of public and private sector organizations in the television industry. Its aim is to establish the framework for the introduction of MPEG-2 based digital television services. Now comprising over 200 organizations from more than 25 countries around the world, DVB fosters market-led systems, which meet the real needs, and economic circumstances, of the consumer electronics and the broadcast industry.

1 Scope

The present document gives guidelines for the implementation of the Generic Stream Encapsulation (GSE) Protocol.

The guidelines are intended to be recommended rules, advice and good practices for the usage of the GSE specifications defined in TS 102 606 [1], in order to assist systems integrators and equipment manufacturers. As such, they aim at facilitating the efficient and reliable implementation of GSE.

Outline of the present document

The present document:

- provides examples of system scenarios where GSE is applicable, with recommendations and advice for using specific GSE features for each scenario;

- provides extensive details about the use and implementation of advanced GSE features, and proposes transmitter and receiver functional architectures;

- addresses the use of the DVB signalling to locate GSE streams;

- presents the physical layer requirements assumed by GSE;

- specifies the use of GSE as an encapsulation protocol for physical layers.

GSE was initially devised for efficiently carrying IP data over DVB-S2 Generic Stream, however it provides a generic encapsulation mechanism potentially applicable to other second generation DVB standards. As a consequence, future revisions may provide new clauses referring to other second generation DVB standards not addressed in the present document.

Updates to the present document will be produced when more results from GSE compliant implementation tests and experience become available.

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific.

- For a specific reference, subsequent revisions do not apply.

- Non-specific reference may be made only to a complete document or a part thereof and only in the following cases:

  - if it is accepted that it will be possible to use all future changes of the referenced document for the purposes of the referring document;

  - for informative references.

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.

2.1 Normative references

The following referenced documents are indispensable for the application of the present document. For dated references, only the edition cited applies. For non-specific references, the latest edition of the referenced document (including any amendments) applies.

[1]   ETSI TS 102 606: "Digital Video Broadcasting (DVB); Generic Stream Encapsulation (GSE) Protocol".

[2]   ETSI EN 302 307: "Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications".

[3]   ETSI EN 301 790: "Digital Video Broadcasting (DVB); Interaction channel for satellite distribution systems".

[4]   ISO/IEC 13818 (parts 1 and 2): "Coding of moving pictures and associated audio".

[5]   ETSI EN 301 192: "Digital Video Broadcasting (DVB); DVB specification for data broadcasting".

[6]   ETSI EN 300 468: "Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB systems".

[7]   IETF RFC 4326 (2005): "Unidirectional Lightweight Encapsulation (ULE) for Transmission of IP Datagrams over an MPEG-2 Transport Stream (TS)", Fairhurst, G. and B. Collini-Nocker.

[8]   IETF RFC 1112 (1989): "Host extensions for IP multicasting", Deering, S STD 5.

[9]   IETF RFC 2464: "Transmission of IPv6 Packets over Ethernet Networks", Crawford, M., December 1998.

[10]   International Journal of Satellite Communications and Networking (2008): "GSE: A Flexible, yet Efficient, Encapsulation for IP over DVB-S2 Continuous Generic Streams", J.Cantillo, B. Collini-Nocker, U. De Bie, O. Del Rio, G. Fairhust, A. Jahn, R. Rinaldo, (GSE Assessment analysis was performed by TriaGnoSis (O.Lücke, A, Jahn) under ESA contract 17403/03).

[11]   IEEE Std. 802.1Q-2005: "Virtual Bridged Local Area Networks".

NOTE: See ISBN 0-7381-3662-X.

[12]   IETF RFC 5163 (2008): "Extension Formats for Unidirectional Lightweight Encapsulation (ULE) and the Generic Stream Encapsulation (GSE)", Fairhurst, G. and B. Collini-Nocker.

[13]   Unidirectional Lightweight Encapsulation (ULE) Next-Header Registry.

NOTE: Available at http://www.iana.org/assignments/ule-next-headers/ule-next-headers.xhtml.

2.2 Informative references

The following referenced documents are not essential to the use of the present document but they assist the user with regard to a particular subject area. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Not applicable.

3 Abbreviations

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

ACM   Adaptive Coding and Modulation
ADSL   Asymmetric Digital Subscriber Line
AF   Assured Forwarding traffic

ATM	Asynchronous Transfer Mode
BB	Base Band
BBF	Base Band Frame
BCH	Bose-Chaudhuri-Hocquenghem multiple error correction binary block code
BE	Best Effort traffic
CCM	Constant Coding and Modulation
CRC	Cyclic Redundancy Check
DFL	Data Field Length
DiffServ	Differentiated Services
DRR	Deficit Round Robin
DSCP	Differentiated Services Code Point
DSL	Digital Subscriber Line
DVB	Digital Video Broadcasting
DVB-S2	Digital Video Broadcasting - Satellite - Second Generation
E	End indicator
EBU	European Broadcasting Union
EF	Expedited Forwarding traffic
EPU	Encapsulated Packet Unit
FEC	Forward Error Correction
FER	Frame Error Rate
FIFO	First In First Out
FragID	Fragmentation Identifier
GARP	Generic Attribute Registration Protocol
GS	Generic Stream
GSE	Generic Stream Encapsulation
GVRP	GARP VLAN Registration Protocol
IANA	Internet Assigned Numbers Authority
ID	Identity
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
INT	IP/MAC Notification Table
IntServ	Integrated Services
IP	Internet Protocol
IPR	Intellectual Property Right
IPTV	Internet Protocol TeleVision
IRD	Integrated Receiver Decoder
ISDN	Integrated Services Digital Network
ISI	Input Stream Identifier
ISO	International Standard Organization
ISP	Internet Service Provider
ISSY	Input Stream Synchronization Indicator
JTC	Joint Technical Committee
LAN	Local Area Network
LDPC	Low Density Parity Check
LT	Label Type indicator
MAC	Medium Access Control
MODCOD	MODulation and CODing
MPE	Multi Protocol Encapsulation
MPEG	Moving Pictures Experts Group
MPLS	Multiple Protocol Label Switching
NCR	Network Clock Reference
NH	Next Header
NIT	Network Information Table
NPA	Network Point of Attachment
NPD	Null Packet Deletion
PDU	Protocol Data Unit
PEP	Performance Enhancing Proxy
PID	Programme Identifier
PL	Physical Layer
POTS	Plain Old Telephone Service
PPPoE	PPP over Ethernet
PS	Professional Services

PSI	Program Specific Information
QoS	Quality of Service
RCS	Return Channel Satellite
RfC	Request for Comment (IETF standard)
RTP	Real-time Transport Protocol
S	Start indicator
SI	Service Information
SNDU	SubNetwork Data Unit
TCI	Tag Control Information
TPID	Tag Protocol ID
TS	Transport Stream
UDP	User Datagram Protocol
ULE	Unidirectional Lightweight Encapsulation
UPL	User Packet Length
VCI	Virtual Channel Identifier
VCM	Variable Coding and Modulation
VDSL	Very high rate Digital Subscriber Line
VLAN	Virtual LAN
VPI	Virtual Path Identifier

4   Requirements for GSE definition

The first generation of DVB standards only supports audio/video transport using the MPEG format [4] with a Transport Stream packet multiplex (MPEG-TS). Multi Protocol Encapsulation (EN 301 192 [5]) is the DVB standard for the encapsulation of data and other content on MPEG-TS packets.

The second generation of DVB standards features backwards compatibility modes for carrying MPEG-TS, as well as generic modes for carrying arbitrary packets of variable length. These are referred to as Generic Streams (GS). Such Generic Streams are intended to transport a sequence of data bits or data packets, possibly organized in frames, but with no specific timing/rate constraints.

GSE was initially devised for DVB-S2 [2] Generic Streams. It was introduced to improve the efficiency of carriage of IP data (and other network and link-layer packets) over DVB-S2 Generic Streams. This clause recalls the requirements which led to the definition of GSE:

NOTE: These requirements were discussed during the 71th DVB-GBS meeting.

- The IP/DVB-S2 protocol shall implement a more efficient encapsulation of IP over DVB-S2 than IP/MPEG-TS using Multi-Protocol-Encapsulation. Given the very large weight of satellite segment cost, IP packets shall be mapped to DVB-S2 frames with minimal overhead. As an objective, for a typical packet length distribution, the target overhead shall be less 3 %. (This overhead includes additional headers for encapsulation and/or segmentation **and** unused portions of DVB-S2 frames (note that In DVB-S2 terminology (EN 302 307 [2] V1.1.1 p18), the losses expressed by the factor DFL/KBCH and by the factor (1-1/UPL) in case of User Packet CRC are taken into account)).

- The protocol shall allow efficient system operation in ACM mode, but shall not mandate a particular ACM forward link scheduling algorithm.

- Considering IP traffic the solution shall cover IPv4 and IPv6. The solution shall also support other types of network traffic, such as PPPoE, IEEE 802.1p/q [11].

- The protocol shall enable the transmission of broadcast, multicast and unicast services.

- The protocol shall not prevent the simultaneous use of 10 000 multicast group destination addresses and 10 000 000 unicast DVB-S2 receivers.

- The protocol shall not prevent the use of Performance Enhancing Proxy (PEP) protocols (Proprietary PEP may be an area in which satellite ISPs compete).

- The protocol shall allow optional broadcast and multicast link encryption on DVB-S2. In encrypted mode, the overhead is allowed to be slightly higher.

- The protocol shall allow simple hardware filtering at the receiver. This shall reduce the software processing in the same way as the PID filtering does in MPEG-TS.

- The maximum packet size shall not be smaller than that imposed by a non-fragmented MPE section, approximately 4 kB.

- The protocol shall have no implication on the ACM scheduler. It is understood that higher layer packets can be fragmented into different BBFrames, where the modulation and coding scheme may change between the BBFrames. Therefore the fragmentation of PDUs in different (and consecutive and non consecutive) BBFrames and the change of MODCOD between fragments shall be supported.

- IP header compression techniques should be supported.

5 System scenarios for GSE applicability

GSE may be used various system scenarios; examples are provided in this clause. For each scenario, if identified meaningful, alternatives to the default GSE addressing format (i.e. IEEE MAC address) are mentioned. For more details about the conditions of application of the GSE multiple addressing formats, see clause 6.1.1.

5.1 Interactive DVB-S2 system with terrestrial return channel

In this scenario, interactive data services including Internet access are provided to consumer Integrated Receiver Decoders (IRD) and to personal computers. Data services are transported over the DVB-S2 [2] forward link in one or multiple Generic Streams using GSE.

Interactivity is established thanks to a terrestrial return link. Both narrowband (3,4 kHz modem or ISDN) as well as broadband (ADSL/VDSL) terrestrial return paths may be considered.

Constant Coding and Modulation (CCM), Variable Coding and Modulation (VCM) or Adaptive Coding and Modulation (ACM) may be provided on the forward link. For ACM, each individual satellite receiving station controls the protection mode of the traffic addressed to it using the terrestrial return channel.
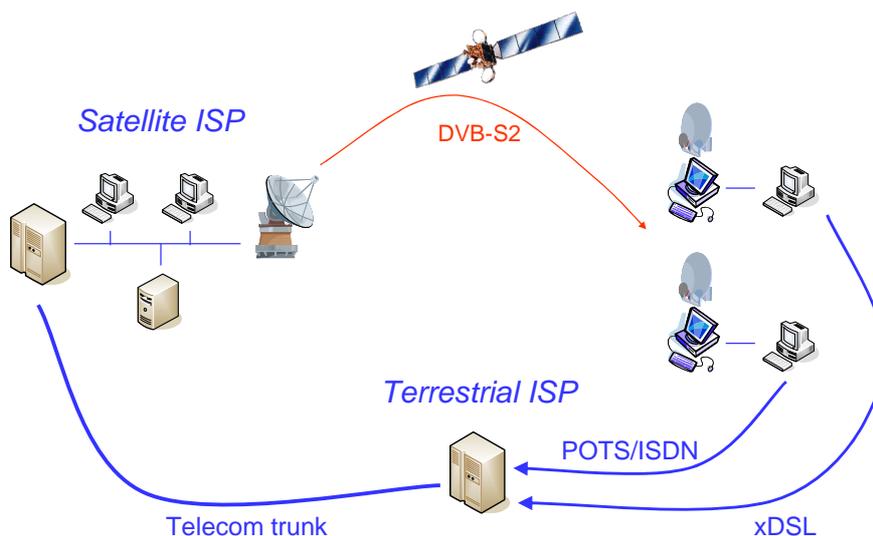

Figure 1: Interactive DVB-S2 system with terrestrial return channel

In this scenario, the default GSE addressing format, i.e. 6-byte label corresponding to the IEEE MAC address, is the most suitable. The "Label re-use" feature can also be used in some cases (c.f. conditions defined in clause 6.1.1.3) to provide overhead reduction.

5.2   Interactive DVB-S2 system with satellite return channel (DVB-RCS) for broadband access

In this scenario, interactive data services including Internet access are provided to consumer Integrated Receiver Decoders (IRD) and to personal computers. It is identical to the first one except that the return IP traffic is carried via the DVB-RCS [3] satellite return channel, using either the DVB-RCS IP/AAL5/ATM or the DVB-RCS IP/MPE/MPEG2-TS encapsulation.

GSE is only employed on the DVB-S2 forward link.

Like in the previous scenario, CCM, VCM or ACM may be provided on the forward link. For ACM, each individual satellite receiving station uses the satellite return channel to control the protection mode of the traffic addressed to it.
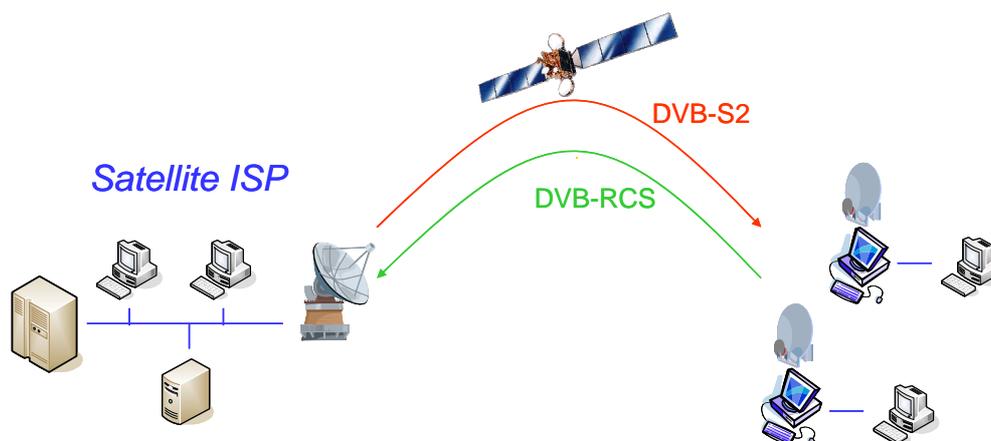

Figure 2: Interactive DVB-S2 system with satellite return channel (DVB-RCS) for broadband access

In this scenario, as an alternative to the 6-byte MAC address label, the 3-byte label may be used, representing either the 1 Byte-group ID and the 2-bytes Login ID assigned to each DVB-RCS terminal, or the VPI/VCI(s) of the ATM connection(s) established on their return link. Furthermore the "Label re-use" feature can also be used in some cases (c.f. conditions defined in clause 6.1.1.3) to provide additional overhead reduction.

5.3   DVB-S2 system for professional applications

This scenario addresses data content distribution/trunking and other professional applications (Professional Services [2]). These services are mainly point-to-point or point-to-multipoint, including interactive services to professional head-ends which re-distribute services over other media.

This scenario is based on a satellite architecture with DVB-S2 [2] in both link directions. That kind of architecture is used to connect major sites that are geographically separated, each network component having a DVB-S2 high-speed link to all others.

Data are transported on each link encapsulated in GSE over a single or multiple Generic Streams.

The system can provide CCM, VCM or ACM.

Figure 3 gives an example of this scenario. A system can be composed of more than 2 stations, with a star or mesh topology, i.e. any station can communicate with any other station, and having unicast, multicast and broadcast communications.
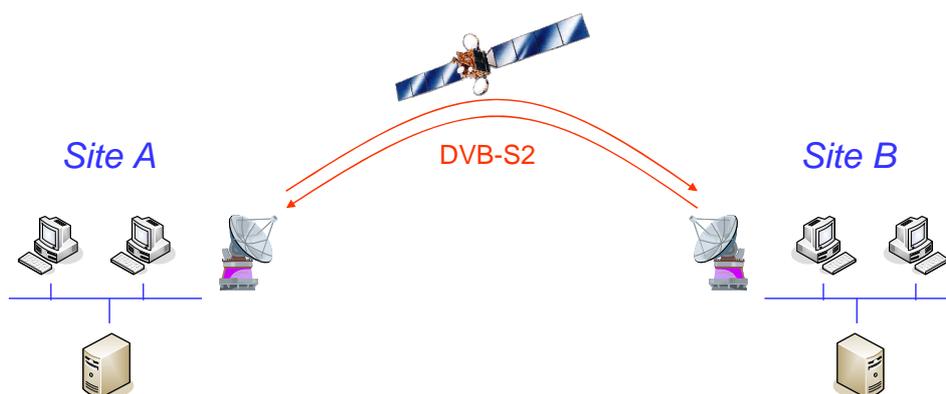
Figure 3: DVB-S2 system for professional applications

Addressing based on 3-byte labels representing "Virtual LANs (VLANs)" may be considered here. The use of "VLAN Labels" is addressed in clause 6.1.1.4. Furthermore, in this type of scenario, each station communicates with few other stations. The "Label re-use" feature can therefore likely be employed very often (c.f. conditions defined in clause 6.1.1.3) to provide additional overhead reduction.

5.4 General broadcasting (unidirectional) of IPTV using DVB-S2 system

This scenario considers the broadcasting of DVB-IPTV services over DVB-S2 satellite links. This scenario can include a terrestrial return channel for interactive TV applications (it is then a particular specialization of the scenario in clause 5.1).

IPTV Data sent over IP is transported encapsulated in GSE packets over one or several Generic Streams (typically there can be one GS per service).

In this system, CCM and VCM may be used.

In this scenario, the "No label" addressing format is a possible alternative to the default GSE addressing format. It may be used for broadcast IPv4 packets. Furthermore the "Label re-use" feature can also be used (c.f. conditions defined in clause 6.1.1.3) to provide additional overhead reduction.

6 GSE implementation

6.1 Implementation and use of advanced GSE features

6.1.1 Multiple addressing format support

GSE supports 4 addressing formats:

- 6-byte label (default format).

- No label.

- Label re-use.

- 3-byte label.

6.1.1.1 Byte label (default format)

The 6-byte label corresponds to the IEEE MAC address (also called Network Point of Attachment) identifying the Receiver(s) of GSE packets. All unicast capable types of terminals must support the corresponding binding.

In case the 6-byte label is used for broadcast IPv4 packets, it shall be set to the NPA link broadcast address (0xFF:FF:FF:FF:FF:FF). Preferably the no-label option should be used for broadcast IPv4 packets.

When the PDU is an IP multicast packet, the IP group destination address of the multicast packet shall be mapped to the multicast GSE Label following the method used to generate a destination MAC address in Ethernet [8], [9] (default method) or any other method which is out of scope of [1] and the present document. The used mapping method is indicated in the DVB signalling, i.e. PSI/SI tables for "Mixed" systems and IP based signalling for "Generic Stream only" systems, c.f. clause 7. In PSI/SI tables, it is signalled in the generic_stream_binding_info conveyed by the selector bytes of the IP/MAC generic_stream_location_descriptor [1]. All multicast capable terminals must support the default multicast binding.

The 6-byte label format is the default label format. All other label formats have been defined to provide optimized solutions reducing overhead but they may be used only in specific conditions.

6.1.1.2 No label

This format (no Label in the GSE Header) should be used when packets will be processed by all Receivers. It must not be used for IP unicast packets delivered to multiple next-hops (i.e. where the same link connects multiple Receivers). It is mainly destined to broadcast IPv4 packets.

Its use may therefore be considered in the scenario in clause 5.4 for broadcast IPv4 packets, as well as in the scenario in clause 5.3 for eventual broadcast IPv4 communications.

This format may also be used if Receivers are able to utilize a discriminator field (e.g. the IPv4/IPv6 destination address or a bridged MAC destination address) of the encapsulated protocol, which could be interpreted as a Layer 2 address. In this case, the "no label" format may be used in any scenario for any type of data (unicast, multicast, broadcast).

All types of terminals must support the "no-label" format, i.e. they must process all packets with this format.

6.1.1.3 Label re-use

If several consecutive PDUs within a same Base Band frame are sent to the same destination, the label re-use feature should be used to reduce overhead: the Label field (a 3-Byte Label or a 6-Byte Label) of all concatenated GSE Packets except the first one can be omitted.

All types of terminals must support the Label re-use format.

The use of this format may be considered in broadcasting scenarios (clause 5.4), however it is less advantageous than the no label format in case of broadcast IPv4 packets.

It may also be considered in the scenario in clause 5.3. Indeed, this kind of systems generally involves few stations, the probability of having consecutive GSE packets with the same destination may be high.

For interactive scenarios (clauses 5.1 and 5.2), Label re-use could be used too. The gain is smaller when the number of Receivers is high.

This feature assumes that GSE encapsulation and GSE packets scheduling in Base band frames are performed jointly; indeed the GSE encapsulator must have the knowledge of which GSE packets are encapsulated in each base band frame.

6.1.1.4 3-Byte label

The 3-byte label may be an alternative to the 6-byte label in scenarios where Label filtering by Receivers is necessary, i.e. IP unicast and multicast packets are sent on links shared by several Receivers. Receiver processing of GSE packets with 3-byte label is optional. Each label must identify a unique receiver or a unique group of receivers in the system.

Three 3-byte label bindings have been defined by the GSE specification:

- DVB-RCS group/logon-ID.

- ATM VPI/VCI.

- VLAN identifier.

If 3-byte labels are used, the binding type is signalled within the PSI/SI in the generic_stream_binding_info conveyed in the selector bytes of the IP/MAC generic_stream_location_descriptor (c.f. clause 7.1). For IP-based signalling, this is not defined yet.

In the scenario in clause 5.2, DVB-RCS terminals may bind their DVB-RCS group/logon-ID to the 3-byte address. This logical address is composed of two fields: the Group_ID and Logon_ID. They are assigned to each terminal during logon. They are used for addressing individual terminals until logoff. The Group_ID corresponds to a group of logged-on terminals. It consists of 8 bits. The Logon_ID uniquely identifies the terminal within a group. The Logon_ID consists of 16 bits. This addressing format is dedicated to point-to-point transmissions (it does not allow to generate multicast labels).

In this scenario, the DVB-RCS return link identifiers, i.e. the VPI/VCI(s) of the ATM connection(s) established on the DVB-RCS terminals' return link, may also be used as 3-byte labels on the forward link. These VPI/VCIs must not be allocated by the network to more than one terminal at any given time if used as 3-byte labels on the forward link. They can be allocated during two different steps: an initial VPI/VCI is transmitted by the network to the terminal after DVB-RCS logon. Then other VPI/VCIs (dedicated to traffic flows) may also be obtained through upper layer signalling (e.g. using the Connection Control Protocol). In case of multiple VPI/VCIs per terminal, an additional signalling mechanism shall be defined to indicate which VPI/VCIs receivers must bind to the 3-Byte address (e.g. the initial VPI/VCI, all etc.) in order they can filter the appropriate labels.

The 3-byte label may be used as well to identify "Virtual LANs" (VLAN).

VLANs with a definition similar to the IEEE 802.1Q [11] definition can be established: multiple bridged networks physically connected to the same DVB stations working as switches share the same DVB physical network link without leakage of information between networks (i.e. trunking). A unique VLAN tag is assigned to each network. The DVB stations encapsulate Ethernet frames using bridged Ethernet frames in GSE packets with the appropriate VLAN tag in the GSE label field (the physical port on which they receive the frame determines the VLAN tag) before broadcasting them on the DVB network. At reception, DVB stations decapsulate all the GSE packets and forward the original Ethernet frames on the appropriate physical port according to the VLAN tag contained in the GSE header.
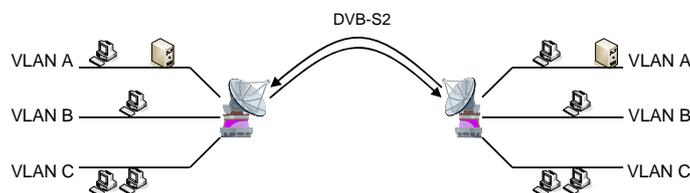

Figure 4: DVB-S2 system for professional application, with IEEE 802.1Q like VLANs

In IEEE 802.1Q [11], the 4-byte VLAN tag is composed of the Tag Protocol ID (TPID), identifying 802.1Q frame followed by the Tag Control Information (TCI) identifying the VLAN. The 802.1Q VLAN tag may be mapped in the GSE 3-byte label by replacing the TPID by a 1-byte field set to "FF".

NOTE: GSE supports the IEEE VLAN specification (as per 802.1Q [11]). The Type is set to 0x8100 (an IEEE assigned value). The Extension Header then becomes the 4B IEEE Tag Protocol (including the VLAN-ID, Priority and payload EtherType). IEEE protocols, such as GVRP, can be used to automate VLAN configurations, and the Tag can (if required) be propagated into Tag-aware Link layer switches, routers and end systems. The format allows the MAC/NPA address to be used (if required) to perform hardware filtering of the packets.

A VLAN may also represent a virtual private subnetwork composed of a restricted number of DVB stations. A GSE packet destined to a virtual LAN contains in its Label field the identifier of the VLAN. Receivers belonging to this VLAN have the appropriate binding and filter the packet. Receivers belonging to another VLAN or which do not belong to any VLAN discard the packet. Professional application scenarios (clause 5.3) can find interest in such a solution, notably when bridge stations are used. Let us consider the scenarioin clause 5.3 (with a mesh DVB-S2/DVB-S2 system with multiple stations). Separate sub-networks or VLANs, composed of distinct groups of stations are established. Each station is configured with the appropriate VLAN ID. Only intra-VLAN communications are allowed. A bridge station encapsulates Ethernet frames received on its terrestrial interface in GSE packets with the Label of its VLAN, before sending them on the satellite link. Receivers belonging to the same VLAN filter the GSE packets thanks to the Label, decapsulate the Ethernet frames, can realize additional filtering upon the Destination Ethernet address and forward the frames on their terrestrial interface.
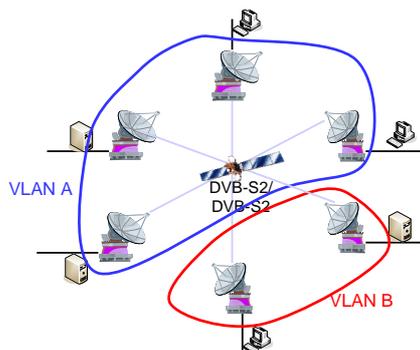
Figure 5: DVB-S2 system for professional application, with VLANs as sub-groups of DVB stations

In order that receivers know how to process VLAN tags or which VLAN IDs are to be filtered, either appropriate configuration must be realized or an additional signalling mechanism must be defined.

6.1.2   GSE Extension Header

Extension Headers offer the opportunity to carry additional protocol fields linked to optional specific functions (such as Link-Layer FEC, Layer 2 security, Compression) without modification of the GSE Header structure.

Extension Header signalling and format defined in RFC 4326 for the Unidirectional Lightweight Encapsulation (ULE) [7], [12] are re-used for GSE.

The presence of an Extension Header in a GSE packet is indicated by the 2-Byte Protocol Type/Extension field of the GSE Header. The set of values that may be assigned to this field follows the rules described in RFC 4326 [7]:

- A Protocol Type/Extension field value less than 1 536 decimal (Type 1) indicates the presence of an Extension Header or identifies a link-specific protocol. The use of Type 1 values must be co-ordinated by an IANA registry [13].

- A Protocol Type/Extension field value greater than or equal to 1 536 (Type 2 - EtherType compatible Type) indicates the protocol type of the PDU (e.g. IPv4, IPv6, Ethernet) carried by the GSE packet. It corresponds to a type code specified by the IEEE/DIX type assignments for Ethernet and recorded in the IANA EtherType registry (e.g. IPv4 payload corresponds to 0x0800).

The Sender may transmit several extension headers in the GSE Extension Headers field. As described in figure 5, they shall be chained in series using a Type field identical to the Protocol Type/Extension field of the GSE Header (with the same rules and semantic for values assignment), indicating the Type of the next header or the encapsulated PDU for the last Extension Header in the packet.

The order of extension headers in GSE packets shall be determined by the fields upon which each extension header operates [12] : firstly extension headers related to link framing and transmission, then extension headers operating on the remaining fields of the GSE packet (consecutive headers and encapsulated PDU), and lastly extension headers linked to the encapsulated PDU.

Implementation shall take into account that a GSE packet may contain several consecutive Extension Headers. The possible maximum number of Extension Headers in a GSE packet will depend on the system characteristics (enabling Link-layer FEC, Layer 2 security, Compression etc.), however it is not expected that GSE packets will generally carry a large number of extensions.

In the case of a PDU fragmented over several GSE packets, Extension Headers (like the Protocol Type and Label fields) are present in the GSE packet containing the first PDU's fragment, as specified in table 2 of [1].
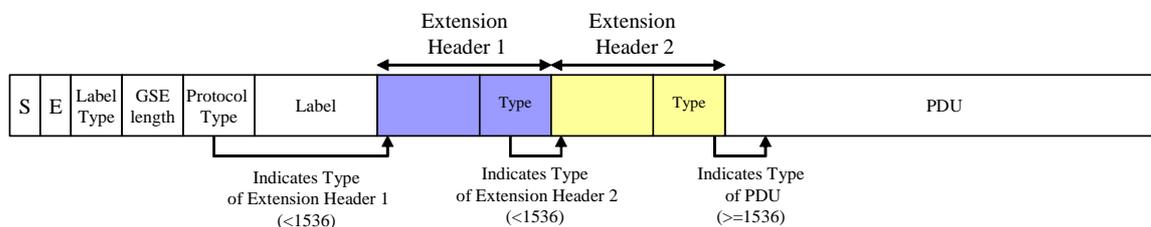

Figure 6: Example of GSE packet including extension headers

For Protocol Type/Extension field Type 1 values (less than 1 536 decimal), GSE uses the semantic specified in [7]. This field is organized as a 5-bit zero prefix, a 3-bit H-Length (H-LEN) field and a 8-bit H-Type field.

0 (5 bits)	H-LEN (3 bits)	H-TYPE (1 byte)

Figure 7: Structure of Protocol Type/Extension field for Type 1 values

The H-Length assignment allows distinguishing between Mandatory and Optional Extension Headers. A H-Length value of zero indicates a Mandatory Extension Header (or link-specific protocol). The length of the Mandatory Extension Header is not communicated in the Protocol Type/Extension field as it has a pre-defined length (and format) that must be known by all GSE receivers. No limit is placed on the maximum length of a Mandatory Extension Header.

A H-Length value of range 1 to 5 indicates an Optional Extension Header, and gives the length of the optional header:

- 1 indicates an Optional Extension Header of length 2 Bytes.

- 2 indicates an Optional Extension Header of length 4 Bytes.

- 3 indicates an Optional Extension Header of length 6 Bytes.

- 4 indicates an Optional Extension Header of length 8 Bytes.

- 5 indicates an Optional Extension Header of length 10 Bytes.

The H-Type is a one-byte field that represents either one of 256 Mandatory Extension Headers or one of 256 Optional Extension Headers.

Thus, IANA Registry values for Mandatory Extension Headers will be decimal numbers less than 256 (decimal). For Optional Extension Headers, they will be decimal numbers in the range 256 to 511 (for the IANA Registry, the H-LEN value is set to 1 whatever the Extension Header length).

Receivers not able to process a Mandatory Extension Header shall drop the PDU. A Mandatory Extension Header may modify the format or encoding of the enclosed PDU (e.g. to perform encryption and/or compression).

Receivers ignorant of an Optional Extension Header shall ignore this header in a GSE packet and process normally all the other fields, i.e. GSE header, other extension headers, encapsulated PDU. This requires that all Optional Extension Headers are defined with the 2-Byte Type field as described in figure 6 (i.e. it must be the last field), and Receivers have the following behaviour:

- They shall determine the length of the unknown extension header and locate its 2-Byte Type field using the H-length value of the Protocol Type field preceding the extension header. The value contained in the 2-Byte Type field will inform them of the type of the next header or PDU. They shall then ignore the unknown extension header and process normally the following GSE packet fields.

6.1.3   Fragmentation Identifier (Frag ID) Management

The Fragmentation Identifier (Frag ID) label is present in the GSE header when the GSE packet encapsulates a PDU fragment. All GSE Packets containing PDU fragments from the same PDU shall contain the same Frag ID. This mechanism supports PDU reassembly even when the GSE Packets containing the fragments of a PDU are interleaved with other GSE packets carrying a full PDU or fragments of other PDUs, addressed to the same or different receivers.

Encapsulator processing:

The sender shall manage the Frag ID label independently for each Generic Stream he generates. Frag ID has 256 possible values (the Frag ID field length is 1 byte). Whenever fragmentation is to be applied to a PDU, the GSE encapsulator shall select a free Frag ID value. GSE Packets containing the fragments of this PDU shall be sent with this Frag ID value and in order. They may be transmitted consecutively, or may be interleaved with other GSE Packets carrying a full PDU or PDU fragments having a different Frag ID. When the sender has completed the transmission of a given PDU, i.e. of all its fragments, the associated Frag ID value may be freed and may then be used for the fragmentation of another PDU. As long as the last fragment is not sent, the Frag ID value must not be freed.

TS 102 606 [1] specifies that if a PDU belonging to a given Frag ID cannot be re-assembled within 255 consecutive base band frames, the Receiver shall discard the already received fragments and free the Frag ID, and this error event should be recorded as a PDU reassembly time-out error. This rule defines an upper bound for the Encapsulator to send the remaining fragment(s) of a PDU. It must send all fragments within the 255 consecutive frames, otherwise the Receiver will consider the PDU as lost and discard any already received fragments.

In systems where GSE packets containing the fragments of a PDU can be sent non-consecutively on a GSE stream, several Frag ID values may be used simultaneously on the same stream: GSE packets carrying the fragments of a PDU may be interleaved with other GSE packets carrying fragments of other PDUs. It is recommended to use as few Frag IDs as possible. An example method is described hereafter to manage properly the set of Frag ID values, i.e. to guarantee that Frag ID values are not used simultaneously for different PDUs: In the GSE encapsulator, for each generic stream, the 256 Frag ID values are stored internally as "available" or "in use". When the encapsulator has to fragment a new PDU, it uses a Frag ID value marked as "available" and changes its description to "in use". When all the GSE packets containing the fragments of this PDU have been encapsulated in Base Band Frames, the Frag ID is set to "available" again.

In systems where GSE packets containing the fragments of a PDU are always sent consecutively on the generic stream, the FragID management may be simpler. Indeed a single Frag ID value may be used all the time. This value is always available for the fragmentation of the following PDU.

Receiver processing:

A Receiver can receive multiple Generic Streams. It must perform PDU reassembly independently for each Generic Stream. To perform the reassembly of a fragmented PDU, the Receiver may use a buffer to hold the partially assembled PDU (implementations may choose other data structures, but shall provide equivalent operations).

Theoretically 256 buffers are required per generic stream, as until 256 Frag ID values may be used at the same time on a generic stream (the maximum number of buffers is therefore the maximum number of generic streams in the system multiplied by 256), however receivers can need less buffers. This depends on the encapsulator scheduler and on its scheduling policy. Three scenarios may be considered, depending on the scheduling policy applied on each generic stream:

- Scenario 1: The Receiver needs only one buffer per generic stream to process, if the GSE scheduler does not allow the interleaving of GSE packets containing PDU fragments belonging to different PDUs (i.e. GSE packets containing the fragments of a PDU are always sent consecutively on the GSE stream).

- Scenario 2: Interleaving of GSE packets containing PDU fragments from different PDUs may happen, however the interleaving of fragments from PDUs destined to the same MAC address never happens in this scenario. The receiver shall implement per generic stream one reassembly buffer for each address present on this generic stream that it binds to or wish to receive (unicast, multicast and broadcast addresses).

- Scenario 3: In all other cases (i.e. interleaving of PDU fragments destined to the same address may happen, different Frag IDs per address may be used simultaneously), a receiver must be able to handle a larger number of concurrently fragmented PDUs up to theoretically a maximum of 256 per generic stream, which requires a larger number of buffers. For instance, in the case where the interleaving of PDU fragments destined to the same address is the result of support for QoS prioritization, the number of buffers per generic stream and for each address present on this generic stream that the receiver binds to or wish to receive, is equal to the number of QoS classes.

In all the above scenarios, a receiver must be able to support multiple Fragment ID values (even when PDUs are all reassembled sequentially using a single reassembly buffer).

When a Receiver receives a GSE packet containing the first fragment of a PDU (S bit set to "1", E bit set to "0"), it checks whether its Frag ID value is already in use, i.e. the Receiver has fragments for that Frag ID in a reassembly buffer. If the Frag ID is already in use, the receiver first discards the already buffered fragments corresponding to this

Frag ID, enters the reassembly process for the given Frag ID and starts reassembly of the new PDU.

If it is not already in use, it starts the reassembly of this PDU, by adding the fragment for this Frag ID to a buffer.

If the GSE Packet contains a continuation or end fragment (S bit is set to "0"), the receiver has to check whether the Frag ID in the GSE header is already in use. If so, it continues with the PDU reassembly processing. Otherwise, the GSE Packet must be discarded.

6.2 Simple transmitter for CCM

This clause describes how a transmitter may perform PDU fragmentation and GSE packet scheduling, when Base Band frames are transmitted with constant transmission parameters (modulation and coding rate) and the GSE scheduler does not provide any support for QoS prioritization. Padding is moreover minimized as much as possible (the cases in which padding is required are described in clause 8.4).

PDU fragmentation and GSE packet scheduling over Base Band Frames may be performed in a straightforward way, as follows:

- Each PDU is fragmented if necessary before being encapsulated in GSE packets, to fit GSE packet length to the remaining space in the current Base Band frame datafield and avoid padding. The only information required by the Fragmentation process from the physical layer is the Base Band frame Data Field Length (which is constant since the transmission parameters are constant).

- GSE packets are inserted consecutively in Base Band Frames since no specific scheduling is required (there is no PDU fragment interleaving).
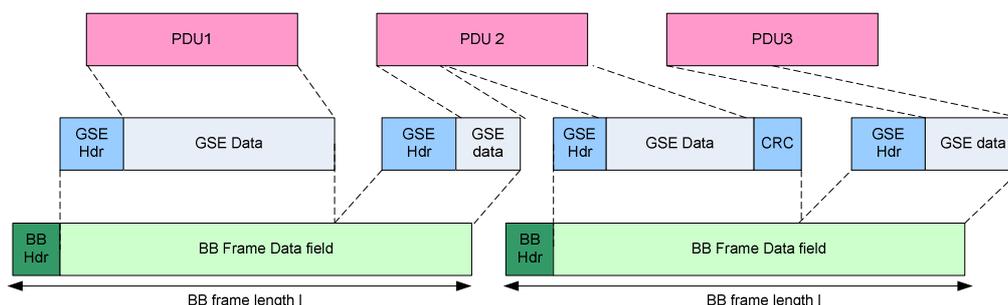

Figure 8: Fragmentation processing with constant transmission parameters

Support for QoS classification and prioritization may be provided by the GSE scheduler. This for instance allows to suspend the transmission of the fragments of a long low-priority PDU and to transmit instead a high-priority PDU. In this case, a specific scheduling is implemented to perform a smart placement of GSE packets in Base band frames taking into account the different classes of services. With this scheduling, the GSE packets containing the fragments of a PDU can be sent in non-consecutive Base Band frames. Mechanisms similar to those described in clause 6.3 may be used to perform this scheduling.

6.3   Transmitter with support for ACM/VCM operation and QoS management

In ACM and VCM modes, transmission parameters (modulation format and coding rate - MODCOD) may vary on a frame-by-frame basis (leading to variable frame sizes).

The flexible PDU fragmentation enabled by GSE allows adapting of the size of each GSE packet to the length of the current Base Band Frame Datafield or to the remaining space in this Datafield. This typically permits a reduction of padding in Base band frames and to optimize capacity gain.

The flexibility of the GSE packets scheduling over Base band frames allows GSE to have no impact on the MODCOD selection for Base band frames: GSE packets containing the fragments of a PDU do not require to be sent in consecutive Base band frames, the selection of the MODCOD of a frame may therefore be independent of the previous frame.

To optimize as much as possible the capacity gain in ACM/VCM context, a smart placement of GSE packets in Base Band frames should be performed. Hence several functions should be performed jointly: PDU fragmentation, GSE encapsulation, GSE packets scheduling and MODCOD selection of Base band frames.

This clause proposes an informative example of a generic implementation for the Generic Stream Encapsulator. This encapsulator processes IPv4 PDUs, is defined to support ACM operations and generates one generic stream. Its high-level functional architecture is described below.

In this example architecture, two topmost units are defined to retrieve, slice, adapt and transmit the incoming PDUs into a proper sequence of GSE packets (called here EPU for Encapsulated Packet Unit). They are the **PDU Manager** and the **EPU Manager**. The joint element between them is a set of **Scheduler Queues** able to store all received PDU packets.
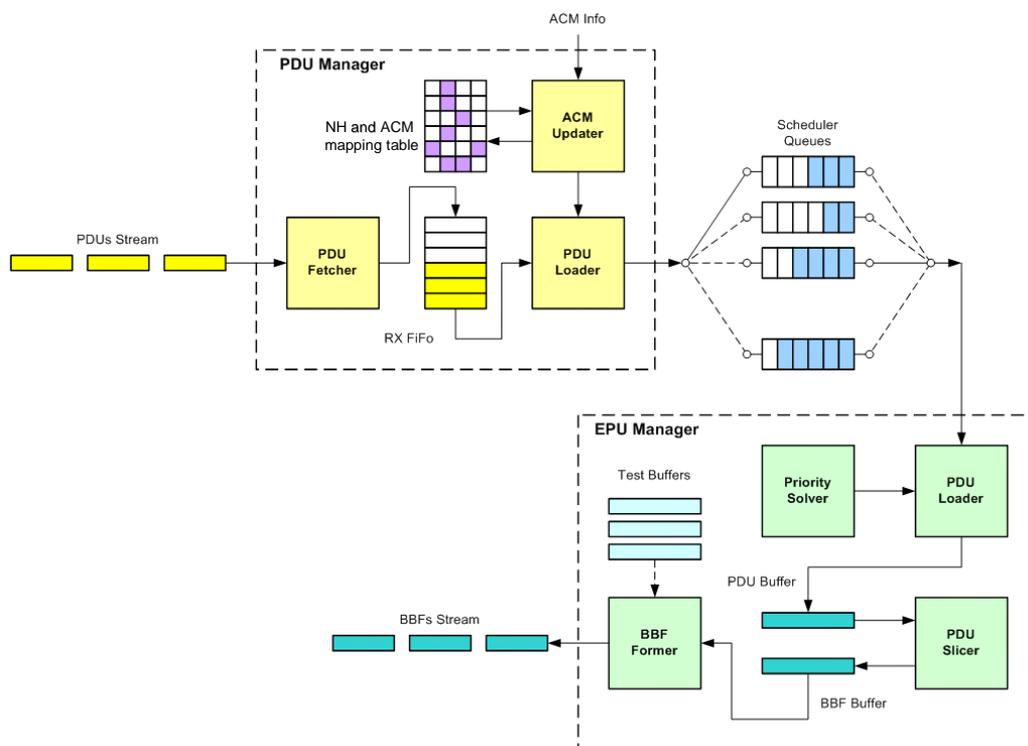

Figure 9: High level scheme of GSE Encapsulator (ACM mode support)

The **PDU Manager** is composed of the following sub-units :

- **PDU Fetcher** - this subunit makes a direct connection with the LAN backbone, loads the incoming packets and stores them inside a RXFIFO.

- **PDU Loader** - the PDU Loader gets recursively packets from the RXFIFO and performs header analysis to check if packets must be processed or not. Three types of conditions have to be satisfied simultaneously to mark the current PDU as "ready for processing":

  - Version matching (IPv4).

  - Protocol matching (TCP, UDP and ICMP check).

  - Destination address vs. Next-hop and ACM mapping table matching (mapping destination IP address to the next-hop neighbour).

- After analysis, each packet is put inside a proper scheduler queue. The queue's choice is made by the **ACM Updater** following the mapping rules that the internal "**Next-Hop and ACM mapping Table**" supplies.

- **Next-Hop and ACM mapping Table** - this internal table gives for each destination IP address the IP address of the corresponding Next-hop neighbour, its GSE Label and the appropriate MODCOD.

- **ACM Updater** - this subunit loads periodically ACM Information (from an ACM Manager) indicating for each next-hop IP address the current transmission conditions, and updates the Next-Hop and ACM mapping table accordingly. In addition, this subunit uses the Destination IP Address that the PDU Loader supplies to perform a key-searching on the Next-hop and ACM mapping table in order to associate to the current packet:

  - The next-hop IP address and GSE label (MAC/NPA address).

  - The appropriate MODCOD.

The **EPU Manager** is composed of the following subunits :

- **PDU Loader** - this subunit retrieves the PDU packets from the Scheduler queues using the criteria that the **Priority Solver** supplies, and inserts them in the PDU Buffer.

- **PDU Slicer** - this subunit fragments PDU packets if necessary and encapsulates them into a suitable sequence of GSE packets that best fits the available base band frame payload.

- **Base Band Frame (BBF) Former** - this subunit retrieves from the base band frame (BBF) Buffer the current frame and finalizes it to make possible its transmission as next base band frame.

Implementations must allow to not send the remaining fragment of a fragmented PDU in the next base band frame (allowing this way that the MODCOD for this frame is independent of the previous frame's MODCOD and that a PDU with higher priority is sent in this new frame). A possible solution for this is to push back the fragment from the PDU buffer in its original FIFO.

Scheduling enforces specific priority rules to select PDU packets to send in the next Base band frame and the MODCOD of this frame. They can be based on the combination of different criteria, e.g. MODCOD efficiency, throughput optimization, etc. They should also take into account the different QoS classes managed by the system (the scheduling should be driven by the system traffic priorities in order not to change the overall Quality of Service).

The example scheduling implementation in the proposed encapsulator is based on a set of scheduler queues reflecting the applied scheduling granularity, i.e. per MODCOD and per QoS, and on the Priority Solver unit, which implements the scheduling algorithm and selects the scheduler queues from which packets will be extracted to fill the base band frames.

The example GSE encapsulator, based on the DiffServ approach, takes into account the following classes of service (the IntServ approach could also be considered):

- Expedited Forwarding traffic (EF).

- Assured Forwarding traffic (AF).

- Best Effort traffic (BE).

The scheduler queues, also called ACM FIFO queues, are thus divided by priority level (EF, AF and BE), and each priority level contains several FIFO queues, each one being associated with a different ACM mode (figure 10).

The Scheduler operations are as follows:

- When a PDU packet comes from the upper layer, it is classified firstly by priority using its Differentiated Services Code Point (DSCP) marking, i.e. QoS, secondly by required protection mode, i.e. MODCOD, and then it is inserted in the corresponding ACM FIFO queue by the **PDU loader.**

- When a base band frame has to be filled for the next transmission, a specific algorithm is applied by the **Priority Solver** to select the ACM FIFO queue from which packets will be extracted to fill the base band frame.

- The protection mode of the base band frame is determined by the MODCOD of the packets selected during these operations.
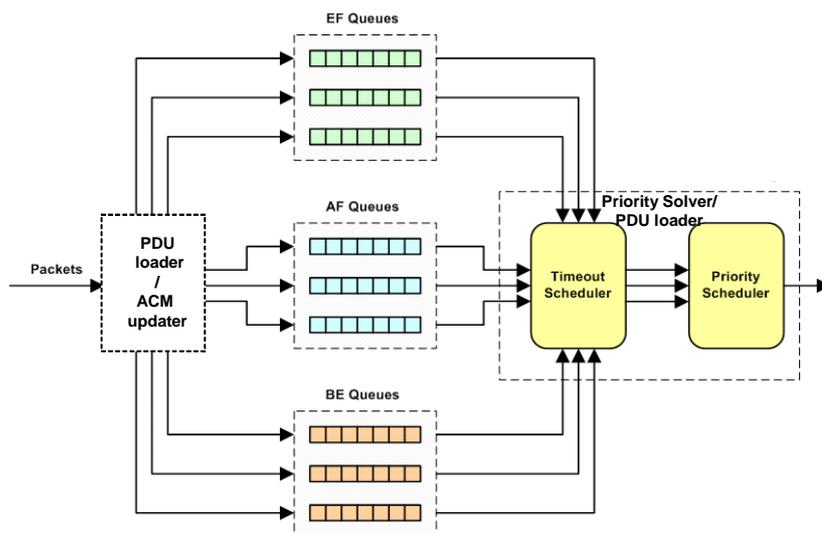

Figure 10: Scheduler queues for ACM and QoS support

NOTE: The logical functionalities "Timeout Scheduler" and "Priority Scheduler" of the Priority Solver are linked to the example priority policy described hereafter.

In the example implementation, if the extracted packets cannot fill completely the base band frame payload, then other packets are fetched from the FIFO queues in order to avoid padding. At this regard it is worth noting that a packet can be aggregated only if it requires a protection level lower or equal to the one of the selected base band frame format. If it is not the case, an alternative solution is to modify the MODCOD and the format of the base band frame according to the protection required by this packet.

Different priority policies may be applied on this example architecture.

The following example is based on the use of timeouts related to the waiting time of the first packet in each FIFO (specific timeouts are implemented for each ACM queue, which are controlled separately by the Timeout scheduler) and on a packet prioritization preferring the highest priority (QoS) packets and among the highest priority packets, packets with the most efficient MODCODs. The algorithm applied when a base band frame has to be filled is described hereafter:

1) The Priority Solver checks if there is one EF packet that is waiting in an EF-ACM queue for a time longer than a certain timeout value. If there is, packets from that EF-ACM queue are selected for the BB-Frame Buffer filling.

2) If no timeout occurs, then the Priority Solver checks if there is one AF packet that is waiting in an AF-ACM queue for a time longer than a certain timeout value (different from the timeout value of EF-ACM queues). If there is, packets from that AF-ACM queue are selected for the BB-Frame Buffer filling.

3) If no timeout occurs, then the Priority Solver checks if there is one BE packet that is waiting in an BE-ACM queue for a time longer than a certain timeout value (different from the timeout value of EF and AF-ACM queues). If there is, packets from that BE-ACM queue are selected for the BB-Frame Buffer filling.

4) If neither of previous conditions is met, then the Priority Solver considers EF-ACM queues and selects packets from the EF-ACM queue having the most efficient MODCOD respect to the least efficient ones.

5) If EF-ACM queues contain no packets, same as point (4) but performed on AF-ACM queues.

6) If AF-ACM queues contain no packets, same as point (4) but performed on BE-ACM queues.

In figure 11, a graphic representation of the priority policy based on the points 4 - 5 - 6 of the algorithm is shown.
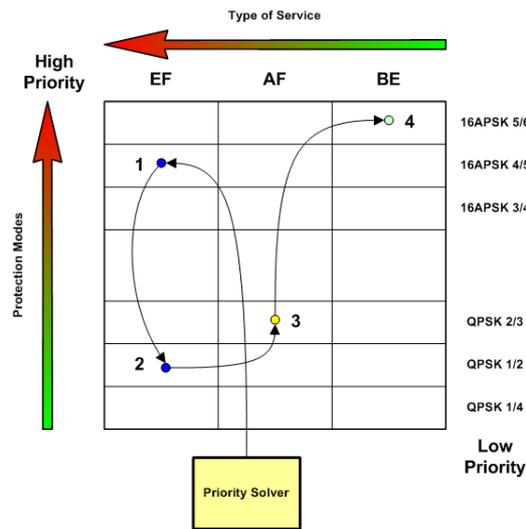

Figure 11: Example of Priority policy for GSE scheduling

Other algorithms may be based on the attribution of a score to each queue, taking into account different criteria such as its QoS priority (EF>AF>BE), its filling level, the latency of first packet to send, etc.

6.4 Receiver

This clause presents an example implementation of the Generic Stream (GSE) Decapsulator (processing one generic stream). Its functional blocks and high-level features are described in figure 12.
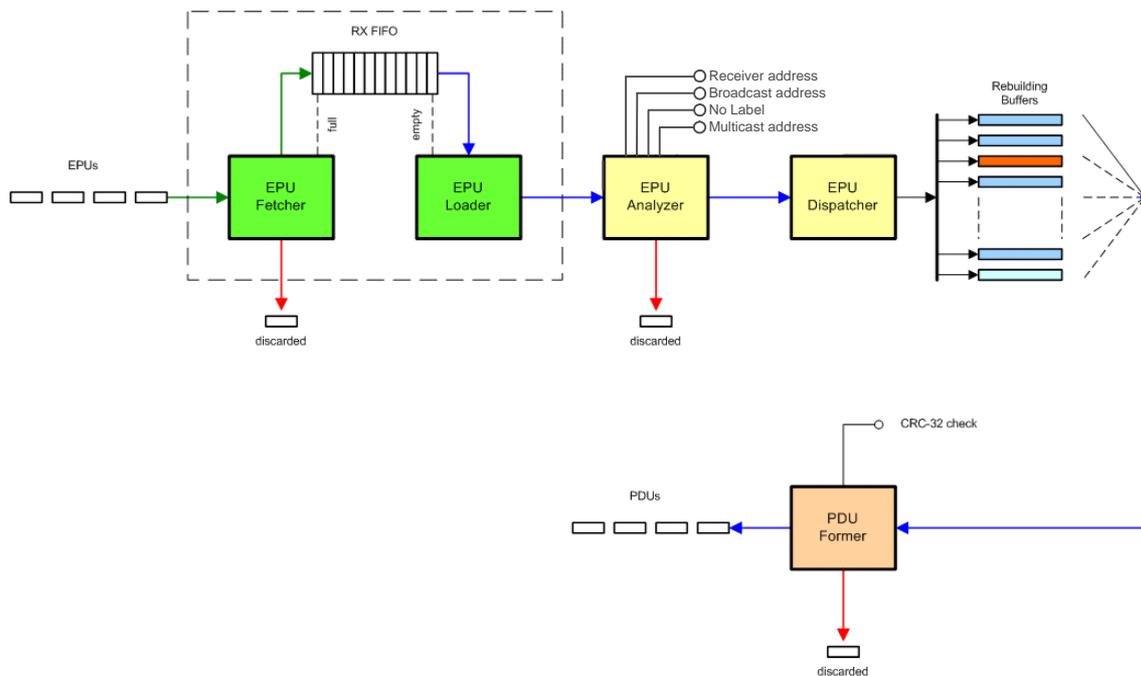

Figure 12: High level scheme of GSE Decapsulator

- **EPU Fetcher** - this subunit is connected to the base band frame Decoder output and loads all the incoming GSE packets. A RXFIFO check is done constantly and when a "Full State" is detected, the EPU Fetcher discards the packets that cannot be stored and processed. An alert is then sent to the decapsulator controller.

- **EPU Loader** - this subunit loads GSE packets and sends them to the EPU Analyzer.

- **EPU Analyzer -** this subunit performs a GSE header analysis in order to recognize if the packet has to be processed or discarded. The following types of filtering are applied:

    - NPA/MAC Receiver matching.

    - Broadcast Address (0xFF:FF:FF:FF:FF:FF).

    - No label.

    - Multicast GSE label.

- **EPU Dispatcher** - this subunit, if the received GSE packet contained a full PDU, directly sends the packet to the PDU. Otherwise it decodes the "Fragment ID" field of the GSE packet and identifies the right "Rebuilding Buffer" where the packet has to be put-in. The EPU Dispatcher uses a buffer structure with 256 elements (as many as the number of Frag ID values). When a GSE packet containing the first fragment of a PDU, i.e. "Start = 1" and "End = 0" is detected, the corresponding "Fragment ID" is stored internally as "in-use". When a GSE packet with the same Frag ID and with "Start = 0" and "End = 1", i.e. containing the last fragment, is received, the buffer content is processed by the PDU Former and the "Fragment ID" internal flag is set to "available" again.

- **PDU Former** - this subunit constantly checks the state of each "Rebuilding Buffer": when a buffer is ready for processing (all fragments of a PDU seem to have been received), the PDU Former realizes the reassembly of the PDU packet structure and checks the PDU using the GSE CRC-32.

6.4.1   Receiver profiles

According to the scenarios defined in clause 5, 3 receiver profiles may be identified:

- Interactive terminal (scenarios in clauses 5.1 and 5.2).

- Terminal for professional/trunking scenarios (scenarios in clause 5.3).

- Basic broadcasting terminal (scenario in clause 5.4).

The GSE Decapsulator architecture described in the previous clause may be mapped into these 3 terminal profiles, the only required adaptations concern the number of rebuilding buffers and the presence of the EPU analyser.

Interactive terminals will typically be used in systems implementing ACM or VCM mode. Optimal transmission parameters will be selected for each transmission frame, depending on criteria related to destination terminal(s) (terminal type and location inside the satellite beam for VCM/ACM, current estimated channel condition for ACM). Support for QoS prioritization may also be performed at GSE level. The GSE scheduler may be implemented so that smart placement of GSE packets in base band frames is performed (i.e. avoiding padding as much as possible). This smart placement may result in the sending of the GSE packets containing the fragments of a PDU in non consecutive frames. As described in clause 6.1.3, terminals must therefore manage several reception buffers, up to 256 per Generic stream to process, to reassemble GSE packets (corresponding to scenarios 2 or 3 described in clause 6.1.3).

Professional/trunking systems may also implement ACM and VCM and provide a support for QoS prioritization at GSE level. Professional/trunking terminals can therefore need to manage many receptions buffers, like Interactive terminals.

In current broadcast scenarios, transmission parameters usually stay constant during all the transmission and are appropriate for all the receivers of the system (data are sent in broadcast). If no support for QoS prioritization is performed at GSE level, the GSE packets containing the fragments of a PDU will be sent in consecutive frames (i.e. no interleaving). This corresponds to the scenario 1 described in clause 6.1.3. Basic broadcasting terminals will need one reception buffer per Generic stream to process. Besides the EPU analyser defined in the 6.4 filtering packets upon their GSE Label is not necessary.

7 Locating GSE streams

7.1 "Mixed" system (TS and GS)

The high-level architecture of "Mixed" systems is described in figure 13. Signalling information (PSI/SI tables) is transported by a Transport Stream (TS). User data are transported over one or several Generic Streams (GS), and possibly over the Transport Stream. The Transport Stream may or may not be multiplexed with the Generic Stream(s). The remaining space within Transport stream base band frames (carrying signalling and possibly data) cannot be used to transport user data from the Generic streams, which can induce bandwidth waste.

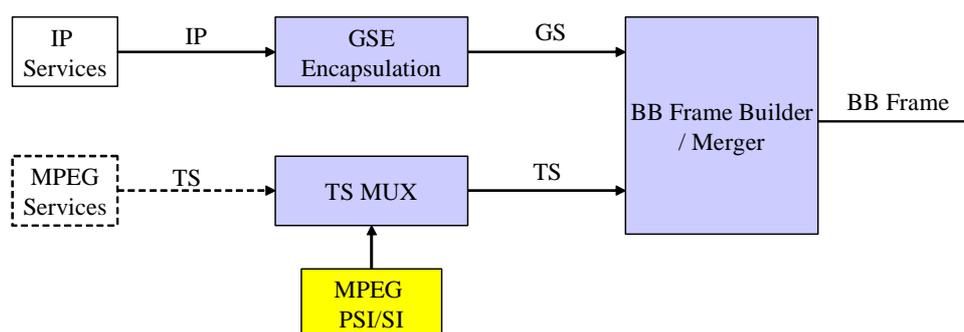"Mixed systems" require receivers to be capable of receiving both Transport and Generic Streams.

Figure 13: "Mixed" system high level architecture

The DVB PSI/SI is used as defined in [5] and [6]. As mentioned in the GSE specification [1], an IP/MAC generic_stream_location descriptor is defined for Generic Streams in TS 301 192 [5]. This descriptor is carried in the IP/MAC Notification Table (INT). The INT table provides a flexible mechanism for carrying the location information (network id, transport id, etc) related to IP/MAC streams within DVB networks. An IP/MAC stream is a data stream including an address header containing an IP and/or MAC address. An IP/MAC stream is encapsulated in a MPEG-2 Transport or Generic Stream multiplex. The INT is divided into sub-tables. Each IP/MAC stream is announced by an INT sub-table.

To obtain the location information related to IP/MAC streams transported over Generic Streams, the Receiver shall listen to the Transport Stream, extract PSI/SI tables and extract from these tables the location information contained in the INT table.
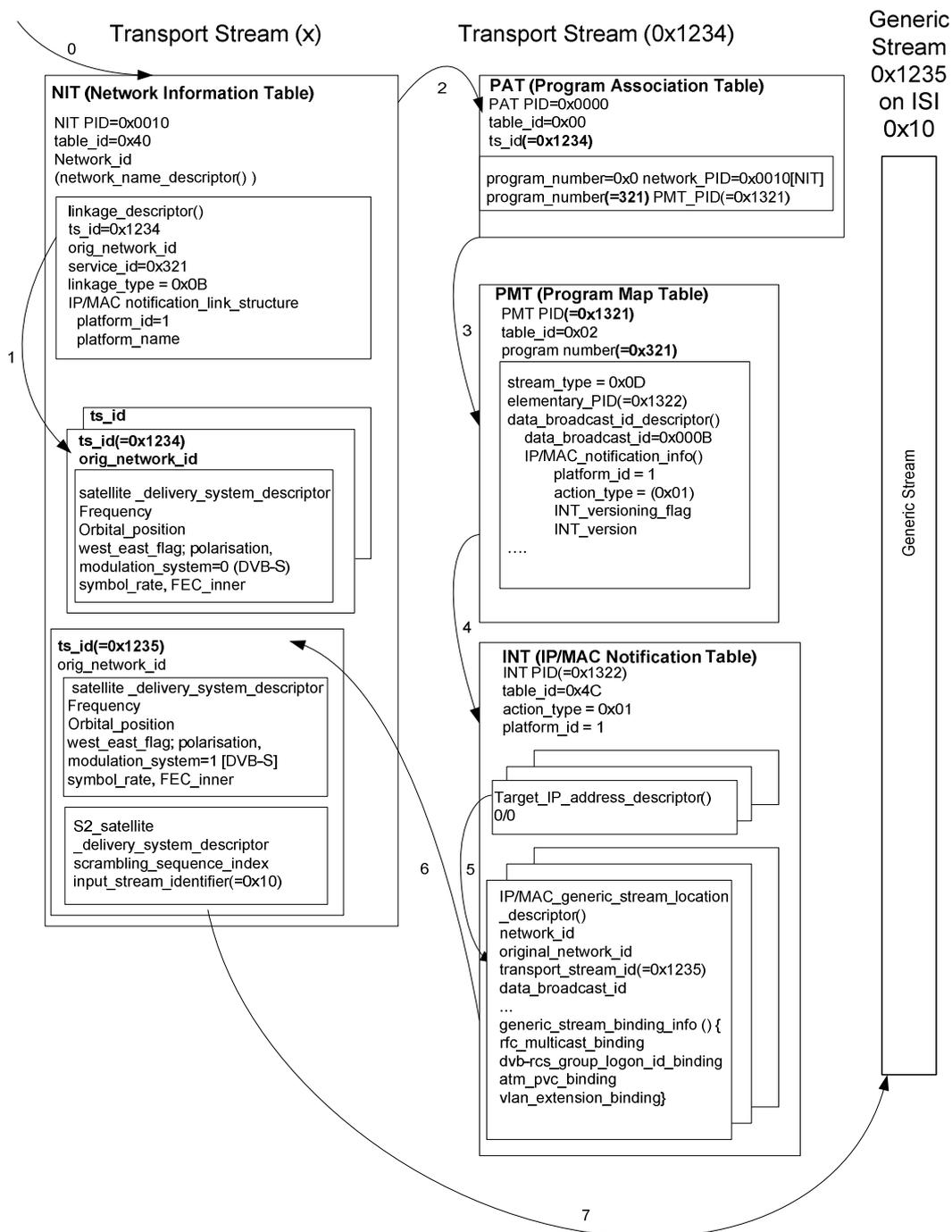
Figure 14: Example for locating a GSE stream in a GS/TS multiplex

The GSE receiver tunes to the start-up transport stream (x) (0) and scans the NIT to locate the transport stream carrying the INT. This must be done by locating the linkage descriptor containing the linkage_type_code 0x0B. If no linkage_type_code 0x0B is found, linkage_type_code 0x0B and linkage_type 0x04 can be searched as described in clause 8.2 "Network (SI) signalling" in DVB-DATA [5] in order to find a transport stream that contains the linkage_type_code 0x0B in the NIT. The receiver must use the ts_id parameter from that linkage descriptor to locate the delivery descriptor for that transport stream (in the second loop of the NIT), and hence the tuning details for the Transport Stream, which carries the INT (1).

The GSE receiver then re-tunes (if necessary) to the Transport Stream that carries the INT(2). The PAT in this transport stream points to a PMT_PID of the PMT for the service_id that was given in the linkage descriptor (3). The PMT of the transport stream carrying the INT contains the data_broadcast_id_descriptor with the data_broadcast_id of 0x000B to indicate the elementary stream used for the IP/MAC Notification table. The GSE receiver then loads the INT on the elementary_PID given in this broadcast_id_descriptor (4).

Then the GSE receiver looks for the INT sub-table section that matches the platform_id and with a target_descriptor_loop that is either empty or that is targeting the GSE receiver (5). The GSE receiver then uses the IP/MAC_generic_stream_location_descriptor from the operational_descriptor_loop, or if that is not present, from the platform_descriptor_loop. This IP/MAC_generic_stream_location_descriptor contains the generic_stream_binding_info to be applied on the generic stream, and the transport_stream_id of the generic stream. With a new lookup in the second loop of the NIT with this transport_stream_id (6), the tuning details of the generic stream are found. In this example the tuning details are contained in the satellite_delivery_system_descriptor and the S2_satellite_delivery_system_descriptor that points to a specific ISI on a DVB-S2 carrier. The GSE receiver finally tunes to this generic stream (7).

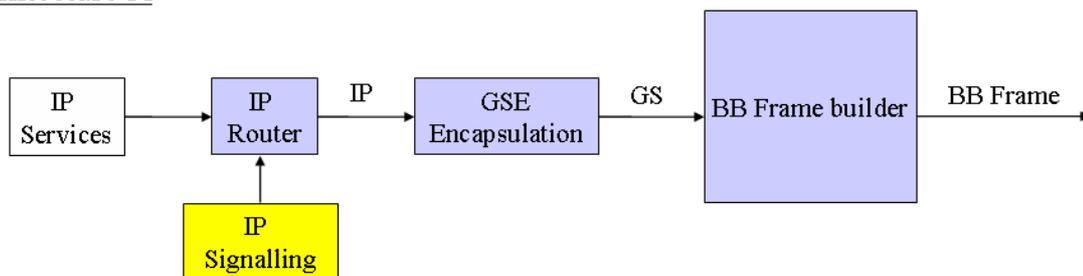# 7.2   "Generic Stream only" system

This type of system is based on one or several Generic Streams. DVB IP Signalling, i.e. signalling based on IP, is transported over a Generic Stream.

Two alternatives may be proposed for the high-level architecture:

- Architecture A : The signalling information and data are transported sharing a common Generic Stream.

- Architecture B : A separate Generic Stream is dedicated to the transport of the signalling information.

The Architecture A is the most efficient regarding capacity optimization, as it allows the remaining space within a signalling BBframe to be used to transport user data.
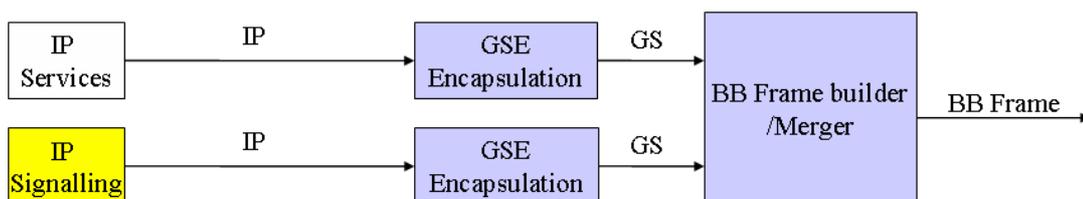

Figure 15: "Generic Stream only" high level architecture

This clause will describe, in a future version of the present document, how to extract the location information related to the Generic Stream(s) of the system in DVB IP signalling. This DVB IP signalling is indeed not yet defined.

In the meantime, an alternative method providing the necessary signalling in "Generic Stream" systems may be used. This method is based on the transport of DVB PSI/SI tables over GSE by using the encapsulation method proposed in clause C.5 of [1]. This method allows the encapsulation of one or several MPEG-TS packets within a GSE packet using a specific protocol type (MPEG-2 TS concatenation protocol type). Architecture A and B may be used. PSI/SI are defined as described in [5] and [6], using the IP/MAC generic_stream_location descriptor for Generic Streams. To obtain the location information related to IP/MAC streams transported over Generic Stream(s), the Receiver must listen to the Generic Stream transporting the PSI/SI tables (in case of multiple generic streams, the receiver shall be configured with the identifier of the GS carrying the signalling), extract PSI/SI tables by filtering GSE packets having the MPEG-2 TS concatenation protocol type, and extract from these tables the location information as described in clause 7.1.

8 Physical layer requirements for GSE applicability

8.1 Reliability requirements

GSE does not include any mechanisms for integrity check of single GSE packets. A CRC-32 is only appended to the last fragment of a fragmented PDU for checking the correctness of the reassembly operation. GSE assumes that the underlying physical/link layer can ensure a sufficient error detection probability.

NOTE 1: A CRC might be added per base band frame at the physical layer level when the FEC channel code cannot ensure the required error detection probability.

NOTE 2: For instance, in DVB-S2 systems, the DVB-S2 FEC code (based on BCH+LDPC) allows Quasi-Error-Free operation: DVB-S2 systems are typically operated at FER=$10^{(-8)}$. This is considered adequate (no additional CRC in DVB-S2 Base Band Frame is necessary).

When a PDU is fragmented and its fragments are scattered over multiple frames, there is a non-negligible probability that a fragment may be missing at receiver side due to transmission errors. The CRC-32 computed at the GSE PDU level for every fragmented PDU allows detection of such a case (which would cause wrong reassembly) with a high probability. It may also detect the presence of uncorrected errors from the physical link, however, as previously mentioned, it is assumed that sufficient physical layer error detection capabilities are present.

For more details about CRC-32 computation and receiver processing, see GSE specifications [1].

8.2 Physical layer Fragmentation

Physical layers may perform fragmentation of GSE packets. This fragmentation shall be transparent to the GSE layer and be defined by the physical layer standard. After processing by the physical layer, GSE PDUs shall be presented to the GSE layer as if no fragmentation had occurred. Physical layer fragmentation may hence be seen as an intermediary sub-layer before the encapsulation of GSE packets and GSE packet fragments in physical layer base band frames (see figure 16).

Thus two fragmentation processes are available: PDU fragmentation performed by GSE (clause 4.2 of [1]) and GSE packet fragmentation performed by the physical layer. One or both fragmentations may be enabled.

All GSE implementations shall implement the PDU fragmentation performed by GSE (clause 4.2 of [1]). This fragmentation can be used in any scenario, and is indispensable for the encapsulation of PDUs larger than 4KB in GSE packets.

The physical layer fragmentation may only be used if it is transparent to the GSE layer.
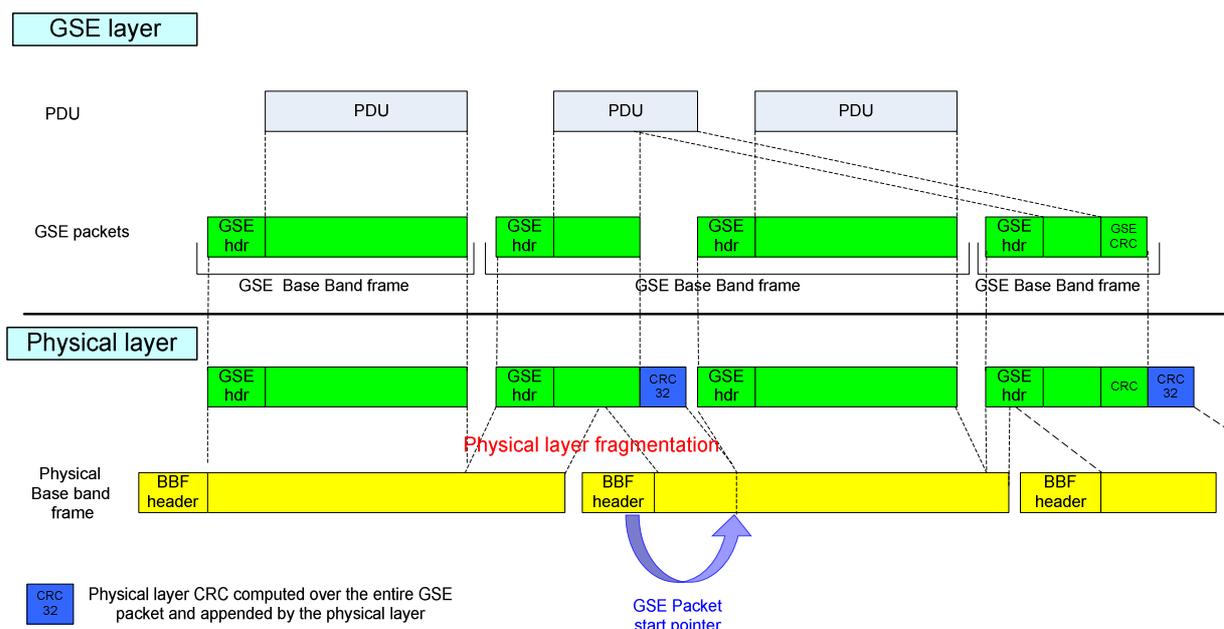
Figure 16: Physical layer fragmentation

The physical layer fragmentation may be performed as described in figure 16. To reassemble GSE packets, the receiver needs to know the start of the first GSE packet in the frame or that no GSE packet starts in the frame. This can for instance be realized as a "GSE Packet Start" pointer in the physical base band frame header (as described in figure 16). It can indicate the beginning of the first GSE packet carried in the frame payload, i.e. the distance in bits from the beginning of the payload to the first GSE header in this frame. A special value of this pointer, i.e. not representing any position in the physical layer payload, may be allocated for the case where no GSE header is present in the payload, i.e. no GSE packet starts in this frame.

NOTE: This "GSE Packet Start" pointer may be similar to the SYNCD field defined by DVB-S2 [2]. This field indicates the distance in bits from the beginning of the data field to the first User Packet bit. It is set to the value 65 535 when no User Packet starts in the data field.

In order to detect reassembly errors at reception, the physical layer should calculate a CRC over the entire GSE packet for every fragmentation case before transmission. The CRC value should be appended to the last fragment of the GSE packet.

The transformation of the fragmented GSE packets from the physical layer to the GSE layer will be done by the reassembly of the original GSE packets and verification of the physical layer fragmentation CRC32. The GSE Base Band frames the GSE specification expects will be reconstructed by starting a new Base Band frame when a new physical layer base band frame starts. In case of a physical layer fragmented GSE packet (fragmented in two or more fragments), the reassembled GSE packet is inserted at the beginning of the next GSE Base band frame.

Thus, on the receiver side, the physical layer shall always present the GSE Base band frames composed of the reassembled GSE packets as a whole and without any possible added CRC, each frame starting with the header of the first GSE packet in the frame.

This fragmentation presented in figure 16 shall not be used in scenarios where fragments of a GSE packet can be sent in non-consecutive base band frames on a generic stream (e.g. in VCM/ACM scenarios, in scenarios with support for QoS prioritization at GSE level). Fragments of a GSE packet shall be sent consecutively on the generic stream, in order to allow reassembly at receiver side.

8.3 Frame format requirements

8.3.1 Start of the first GSE packet

Data frames presented to the GSE layer upon reception shall always start with the header of the first GSE packet.

8.3.2   Frame size requirements

GSE does not work with frame size of less than the maximum used label size (0,3 bytes or 6 bytes) + 7 bytes. This corresponds to the smallest GSE packet carrying the start of a PDU, i.e. Start Indicator set to "1"and End indicator set to "0".

8.4   Padding

If padding is performed at physical layer level, all padding bits shall be set to zero. The length of the useful data within the physical layer frame may be conveyed by signalling at physical layer level.

If information on the actual data occupation of the physical layer frame is not available, the receiver shall recognize the presence of padding (for discarding it) through detection of a specific combination of the Start Indicator, End Indicator and Label Type indicator bits: Start Indicator and End Indicator set to "0", and Label Type Indicator is set to "00". This yields a series of 4 consecutive zero bits. Hence the requirement that all padding bits shall be set to zero.

Padding is used in the following cases:

- If no Layer 3 data (with an appropriate MODCOD for ACM) is available to fill the remaining space of the current base band frame (rare case).

- If the remaining space of the base band frame datafield is smaller than the smallest GSE packet (the physical layer does not implement the fragmentation presented in clause 8.2). The length of the smallest GSE packet (according to [1]) is always less than 14 bytes and is case dependant:

  - For a GSE packet carrying a full PDU (Start and End indicators set to "1"): 10-byte GSE header (6-byte label, no extension header), assuming no data.

  - For a GSE packet carrying a first PDU fragment (Start Indicator set to "1"and End indicator set to "0"): 13-bytes GSE header (6-byte label, no extension header), assuming no data.

  - For a GSE packet carrying a mid PDU fragment (Start and End indicators set to "0"): 3-byte GSE header, assuming no data.

  - For a GSE packet carrying an end PDU fragment (Start Indicator set to "0"and End indicator set to "1"): 3-byte GSE header + 4-bytes CRC, assuming no data.

  NOTE 1: The GSE Length field in the GSE Header indicates the length, in bytes, of the GSE Packet counted from the byte following this GSE Length field. Its value will therefore be always different from 0. For the smallest packet (Start and End indicators set to "0"), the GSE Length is equal to 1.

  NOTE 2 : For the cases Start and End indicators set to "1" and Start and End indicators set to "0", the transmission of the smallest packets does not make much sense; transmission of padding instead may therefore be considered.

- If the encapsulator, when reaching the end of a base band frame, chooses to pad the frame instead to send a partial PDU.

8.5   GSE packets no re-ordering

GSE packets shall not be re-ordered between encapsulator and decapsulator, as specified in [1].

8.6   Signalling requirements

The physical layer should signal the type of encapsulation used, e.g. GSE, when this is not predetermined by system configuration.

9   DVB-S2 systems

The DVB-S2 standard is specified in EN 302 307 [2]. GSE was initially devised for efficiently carrying IP data over DVB-S2 Generic Stream.

9.1   Reliability

The DVB-S2 standard is characterized by a powerful FEC system based on LDPC (Low Density Parity Check) codes concatenated with BCH codes, allowing Quasi-Error-Free operation. DVB-S2 systems are typically operated at FER=$10^{(-8)}$. DVB-S2 FEC code (BCH+LDPC) allows an overall frame error non-detection probability lower than $10^{(-12)}$.
This is considered adequate compared to the requirements of GSE (c.f. clause 8.1) ; no CRC is necessary at physical layer level.

9.2   Frame format

Figure 17 describes the format of DVB-S2 Base Band Frame encapsulating GSE packets.
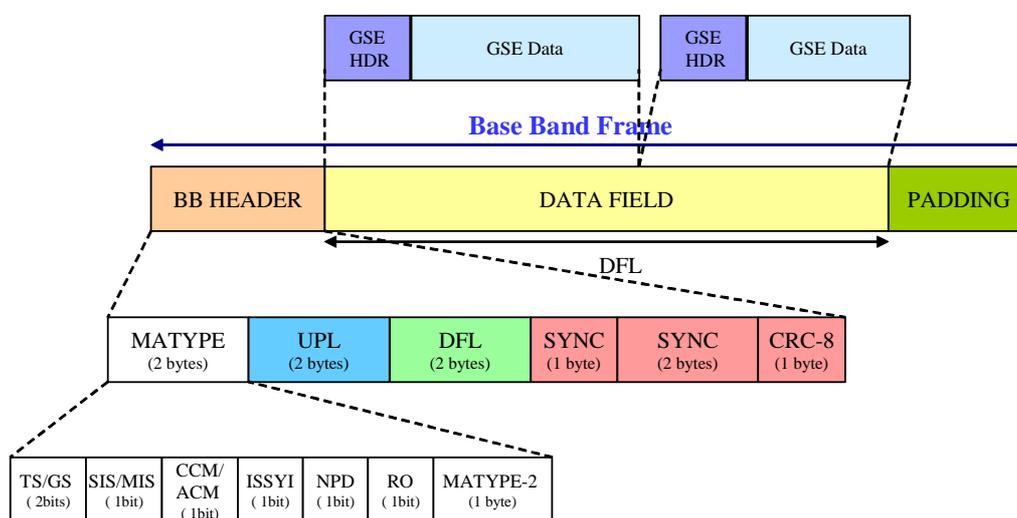

Figure 17: DVB-S2 Base Band Frame format

In DVB-S2, GSE packets are not fragmented between Base Band (BB) frames, therefore a DVB-S2 Base Band frame always starts with the header of the first GSE packet.

If padding is required to completely fill a BB frame, the necessary number of zero bits is appended after the Data Field, and the information related to the length of the useful data within the BB frame is provided by the Data Field Length (DFL) in the BB Header.

In DVB-S2, GSE packets are not expected to be re-ordered between encapsulator and decapsulator.

GSE packets are carried over Generic Continuous Streams.

When the BB Frame contains GSE packets, some BB Header fields contain specific values or are not defined:

- The Input Stream Synchronization Indicator (ISSY) field (1 bit) must be set to 0 (= inactive). The function associated with this field allows guaranteeing a constant-bit-rate and therefore is used only for packetized input streams.

- The Null-Packet Deletion (NPD) field (1 bit) must be set to 0 (= inactive). Indeed the associated function aims at identifying and removing MPEG null-packets. It is therefore not relevant for GSE packets.

- The User Packet Length (UPL field) (2 bytes) must contain the value 0000HEX for continuous stream.

- The SYNCD field is not defined for Generic Streams (this field, which indicates the distance in bits from the beginning of the Data field and the first User Packet of the frame, is defined only for Transport Streams in the DVB-S2 standard).

- The value of the TS/GS and SYNC fields is described in clause 9.4.

9.3   Input Stream Identifier (ISI)

In DVB-S2, multiple streams may be multiplexed at the transmitter. Generic Stream Encapsulation (including fragmentation) shall be carried out separately for the incoming data of each generic stream. Each stream is identified by a specific Input Stream Identifier (ISI). Its value is present in the BB header (in the second byte of the MATYPE field) in case of multiple input streams.

The ISI of each (generic or transport) stream in DVB-S2 is signalled in the S2 satellite delivery system descriptor [6] describing the stream. This descriptor is located in the Network Information Table (NIT).

9.4   Signalling

In DVB-S2, Generic Continuous Stream is signalled in the BB Header of DVB-S2 frames by the TS/GS field (belonging to the first Byte of the MATYPE field), with a value of 01.

For Generic Continuous Streams, the value of the SYNC field is used for transport layer protocol signalling. For GSE, the SYNC field value is as defined in [2].

Annex A (normative):
GSE performances over DVB-S2 satellite links

This annex presents a performance analysis for GSE over DVB-S2 generic streams. This analysis [10] was based on simulations where the different components of DVB-S2 systems in Adaptive Coding and Modulation (ACM) mode were considered, as well as realistic traffic and fading conditions. Both GSE and MPE/MPEG-TS encapsulation were simulated in order to assess the GSE efficiency.

Clause A.1 presents the different assumptions of the analysis; clause A.2 presents the results.
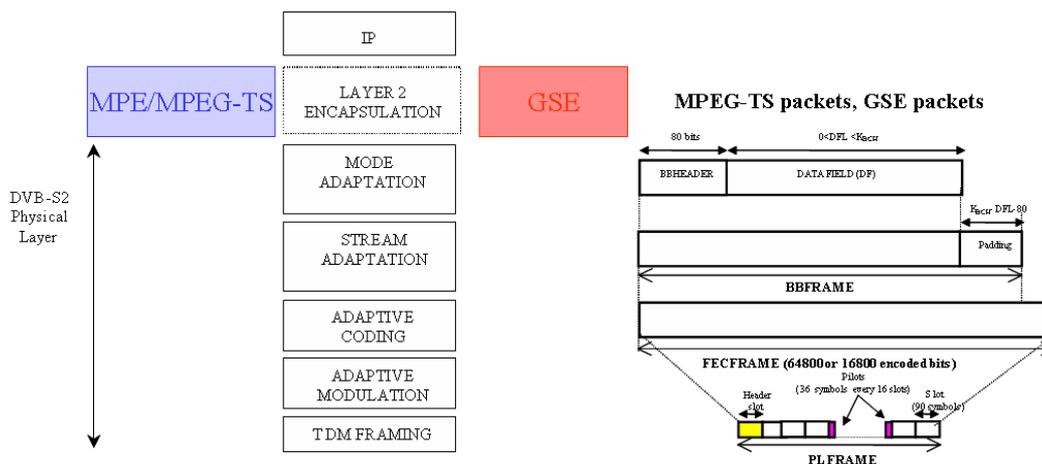

Figure A.1: Location of MPE/MPEG-TS and GSE encapsulations in the DVB-S2 stack

A.1 Assessment assumptions and scenarios

The simulator structure, based on a possible DVB-S2 gateway, is described in figure A.2.

The Traffic Generator produces IP datagrams with arrival time and length statistics according to some well defined traffic models (described hereafter). Each traffic flow is associated with a receiver terminal, to which the Protection Level Control associates dynamically a minimum protection level (MODCOD) depending on the fading situation (it changes throughout the simulation period).

The MODCOD Classifier forwards each IP datagram it receives from the Traffic Generator to the corresponding MODCOD queue.

The Deficit Round Robin (DRR) scheduler periodically selects a MODCOD queue for transmission. The DDR weights are calculated proportionally to the MODCOD buffer length, and are updated after each Base Band Frame transmission. PDUs are taken from the selected MODCOD queue and are encapsulated in GSE packets, which are then placed in the Base Band Frame datafield. This process is repeated until the Datafield is completely filled. Any unsent PDU or PDU fragment remain in the MODCOD queue.
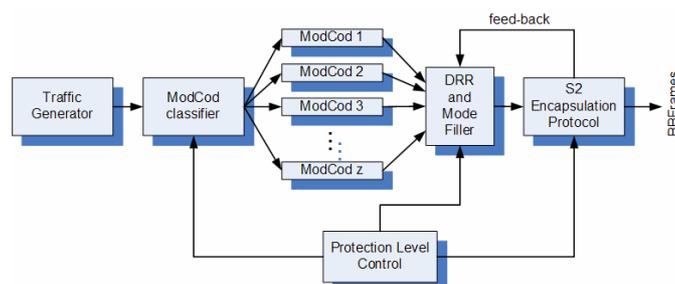

Figure A.2: Simulator structure

The features allowed by the GSE encapsulation/fragmentation are implemented: PDU fragments like entire PDU can be sent in any Base Band frame with a compatible MODCOD (with same or higher protection), and fragments may be placed arbitrary in the Base Band frame. The scheduler is able to take additional packets from MODCOD queues with lower protection level, in case the Datafield of a Base Band frame to be sent with a given MODCOD cannot be fully filled. It waits first a threshold time for the arrival of further PDUs and if this timer expires, it fills the remaining bytes with PDUs requiring less protection.

For the MPE/MPEG-TS implementation, the most favourable configuration for MPE, i.e. with section packing, is implemented in the simulator. Moreover, non-consecutive fragmentation across Base Band frames is supported. This is carried out by assigning a dedicated PID to each MODCOD. All fragments of the same MPE packet are sent consecutively with respect to the MPEG cells having the same PID, and it is not allowed to use PDU fragments to fill a BBframe having a different MODCOD (i.e. fragments must be sent with the MODCOD associated with their MODCOD queue).

The performances assessment was conducted under realistic traffic and satellite channel conditions. For traffic, a mix of HTTP, e-mail, VoIP and MPEG signalling was assumed. In addition a scenario with HTTP-only traffic was considered.

Table A.1: Traffic mix

Traffic	HTTP	Email	VoIP	MPEG
% of total traffic	55 %	15 %	20 %	10 %

The channel model was based on a fading event simulation representing a typical rain fade event for Western Europe. The channel model was used to derive a set of time series identifying during a period of 30 minutes the minimum required MODCOD values for 500 terminal locations (the terminals are located in one spotbeam, focusing on France, Benelux and Germany; they are spread in an area of 500 km x 500 km). The subset of MODCOD modes used for this analysis is mentioned in table A.2.

Table A.2: Used ACM MODCOD modes (Efficiencies are given for normal FECFRAME length and no pilots)

MODCOD	Efficiency (bit/symbol)	MODCOD	Efficiency (bit/symbol)
QPSK 1/4	0,49	QPSK 4/5	1,59
QPSK 1/3	0,66	QPSK 5/6	1,65
QPSK 2/5	0,79	8PSK 3/5	1,78
QPSK 1/2	0,99	8PSK 2/3	1,98
QPSK 3/5	1,19	8PSK 3/4	2,23
QPSK 2/3	1,32	8PSK 5/6	2,48
QPSK3/4	1,49	16APSK 2/3	2,64

The performance assessment was conducted for normal FECFRAME.

A.2   Performances results

Both overhead induced by GSE and MPE/MPEG-TS and the Information transmission efficiency were evaluated.

The overhead was defined as:

- Overhead factor $[\%] = \Sigma_{\text{all packets}} (\text{overhead\_size}) / \Sigma_{\text{all packets}} (\text{overhead\_size} + \text{PDU size}) \times 100$ [%].

Where overhead_size includes with GSE, the GSE encapsulation header and CRC when fragmentation is used, and with MPE/MPEG-TS, the MPE encapsulation header and CRC and the effect of the 4-Byte MPEG-TS packet header.

Table A.3: Simulated overhead for GSE and MPE/MPEG-TS

	HTTP only	Traffic Mix
GSE	2,3 %	4,9 %
MPE/MPEG-TS (packing)	9,5 %	13,8 %

According to the results presented in table A.3, GSE reduces the transmission overhead compared to MPE by approximately a factor of 3.

Figure A.3 describes the GSE overhead distribution for the traffic mix scenario. For more than 90 % of PDUs, the number of overhead bytes is 10 bytes. This means that more than 90 % of PDUs are not fragmented by the GSE protocol.
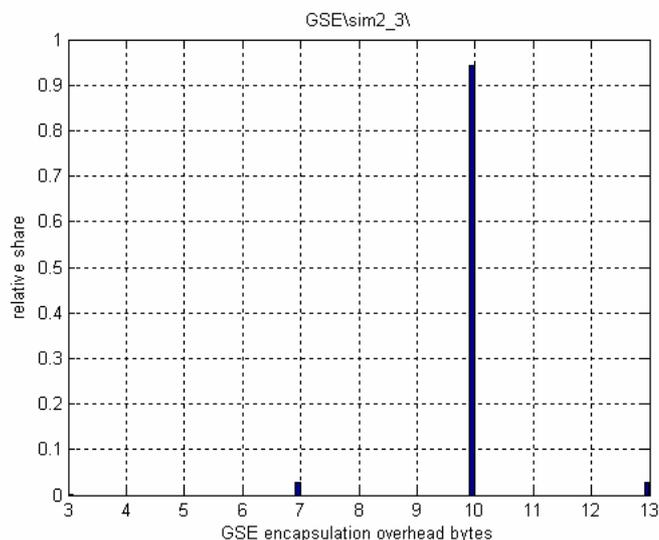

Figure A.3: GSE overhead distribution (for traffic mix)

The Information transmission efficiency achieved by the system was defined by:

- Information transmission efficiency [bit/symbol] = $\sum_{\text{all frames}}$ (transmitted information bits per BBFrame) / $\sum_{\text{all frames}}$ (PLFrame length in symbols - PLHeader length in symbols).

Table A.4: Simulated Information transmission efficiency for GSE and MPE/MPEG-TS[bits/symbol]

	HTTP only	**Traffic Mix**
GSE	1,1 bits/symbol	1,78 bits/symbol
MPE/MPEG-TS (packing)	1,19 bits/symbol	1,95 bits/symbol

According to the results presented in table A.4, GSE leads to an improvement of about 10 % in throughput.

History

Document history		
V1.1.1	June 2009	Publication