The Consultative Committee for Space Data Systems

Recommendation for Space Data System Standards

ENCAPSULATION PACKET PROTOCOL

Proposed Changes to CCSDS 133.1-B-2 BLUE BOOK

Editing date as per file name below

Encap 133x1b2c2_20190322-GPCallAccepted.docx

AUTHORITY

Issue:	Recommended Standard, Issue 2
Date:	October 2009
Location:	Washington, DC, USA

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and represents the consensus technical agreement of the participating CCSDS Member Agencies. The procedure for review and authorization of CCSDS documents is detailed in the *Procedures Manual for the Consultative Committee for Space Data Systems*, and the record of Agency participation in the authorization of this document can be obtained from the CCSDS Secretariat at the address below.

This document is published and maintained by:

CCSDS Secretariat
Space Communications and Navigation Office, 7L70
Space Operations Mission Directorate
NASA Headquarters
Washington, DC 20546-0001, USA

STATEMENT OF INTENT

The Consultative Committee for Space Data Systems (CCSDS) is an organization officially established by the management of its members. The Committee meets periodically to address data systems problems that are common to all participants, and to formulate sound technical solutions to these problems. Inasmuch as participation in the CCSDS is completely voluntary, the results of Committee actions are termed **Recommended Standards** and are not considered binding on any Agency.

This **Recommended Standard** is issued by, and represents the consensus of, the CCSDS members. Endorsement of this **Recommendation** is entirely voluntary. Endorsement, however, indicates the following understandings:

- Whenever a member establishes a CCSDS-related **standard**, this **standard** will be in accord with the relevant **Recommended Standard**. Establishing such a **standard** does not preclude other provisions which a member may develop.
- Whenever a member establishes a CCSDS-related **standard**, that member will provide other CCSDS members with the following information:
  - -- The **standard** itself.
  - -- The anticipated date of initial operational capability.
  - -- The anticipated duration of operational service.
- Specific service arrangements shall be made via memoranda of agreement. Neither this **Recommended Standard** nor any ensuing **standard** is a substitute for a memorandum of agreement.

No later than five years from its date of issuance, this **Recommended Standard** will be reviewed by the CCSDS to determine whether it should: (1) remain in effect without change; (2) be changed to reflect the impact of new technologies, new requirements, or new directions; or (3) be retired or canceled.

In those instances when a new version of a **Recommended Standard** is issued, existing CCSDS-related member standards and implementations are not negated or deemed to be non-CCSDS compatible. It is the responsibility of each member to determine when such standards or implementations are to be modified. Each member is, however, strongly encouraged to direct planning for its new standards and implementations towards the later version of the Recommended Standard.

FOREWORD

This document is a **Recommended Standard** for use in developing flight and ground systems for space missions and has been prepared by the **Consultative Committee for Space Data Systems** (CCSDS). The Encapsulation Service described herein is intended for missions that are cross-supported between Agencies of the CCSDS.

This **Recommended Standard** specifies a communications service to be used by space missions to transfer protocol data units that are not directly transferred by the Space Data Link Protocols (references [1]-[4]) over a ground-to-space or space-to-space communications link. The data units transferred with this service are encapsulated in Encapsulation Packets, defined in this document.

This **Recommended Standard** is developed from the Encapsulation Service that was defined in the Advanced Orbiting Systems (AOS) Recommended Standard (reference [B2]). In this **Recommended Standard**, that service is re-defined so that it can be used with any of the Space Data Link Protocols (references [1]-[5]). Also, the Encapsulation Packet that was defined in references [B2]-[B4] is included in this **Recommended Standard** as an alternative packet structure for encapsulation.

Through the process of normal evolution, it is expected that expansion, deletion or modification to this document may occur. This Recommended Standard is therefore subject to CCSDS document management and change control procedures, as defined in reference [B1]. Current versions of CCSDS documents are maintained at the CCSDS Web site:

http://www.ccsds.org/

Questions relating to the contents or status of this document should be addressed to the CCSDS Secretariat at the address indicated on page i.

At time of publication, the active Member and Observer Agencies of the CCSDS were:

Member Agencies

- Agenzia Spaziale Italiana (ASI)/Italy.
- British National Space Centre (BNSC)/United Kingdom.
- Canadian Space Agency (CSA)/Canada.
- Centre National d'Etudes Spatiales (CNES)/France.
- China National Space Administration (CNSA)/People's Republic of China.
- Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR)/Germany.
- European Space Agency (ESA)/Europe.
- Russian Federal Space Agency (RFSA)/Russian Federation.
- Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
- Japan Aerospace Exploration Agency (JAXA)/Japan.
- National Aeronautics and Space Administration (NASA)/USA.

Observer Agencies

- Austrian Space Agency (ASA)/Austria.
- Belgian Federal Science Policy Office (BFSPO)/Belgium.
- Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
- Centro Tecnico Aeroespacial (CTA)/Brazil.
- Chinese Academy of Sciences (CAS)/China.
- Chinese Academy of Space Technology (CAST)/China.
- Commonwealth Scientific and Industrial Research Organization (CSIRO)/Australia.
- CSIR Satellite Applications Centre (CSIR)/Republic of South Africa.
- Danish National Space Center (DNSC)/Denmark.
- European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
- European Telecommunications Satellite Organization (EUTELSAT)/Europe.
- Geo-Informatics and Space Technology Development Agency (GISTDA)/Thailand.
- Hellenic National Space Committee (HNSC)/Greece.
- Indian Space Research Organization (ISRO)/India.
- Institute of Space Research (IKI)/Russian Federation.
- KFKI Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- Korea Aerospace Research Institute (KARI)/Korea.
- Ministry of Communications (MOC)/Israel.
- National Institute of Information and Communications Technology (NICT)/Japan.
- National Oceanic and Atmospheric Administration (NOAA)/USA.
- National Space Organization (NSPO)/Chinese Taipei.
- Naval Center for Space Technology (NCST)/USA.
- Scientific and Technological Research Council of Turkey (TUBITAK)/Turkey.
- Space and Upper Atmosphere Research Commission (SUPARCO)/Pakistan.
- Swedish Space Corporation (SSC)/Sweden.
- United States Geological Survey (USGS)/USA.

DOCUMENT CONTROL

Document	Title and Issue	Date	Status
CCSDS 133.1-B-1	Encapsulation Service, Recommended Standard, Issue 1	June 2006	Original issue, superseded
CCSDS 133.1-B-2	Encapsulation Service, Recommended Standard, Issue 2	October 2009	Current issue: – adds corrections and clarifications to the specification
CCSDS 133.1-B-2 Cor.1	Technical Corrigendum 1	September 2012	Replaces reference to CCSDS 135.0-B-4, *Space Link Identifiers*, with reference to SANA.
CCSDS 133.1-B-2 Cor.2 EC 1	Technical Corrigendum 2 Editorial change 1	April 2014	Cor.2: – replaces reference to SANA with direct references to SANA registries; – expands range of reserved Space Packet APIDs to include CFDP. EC 1: – updates superseded references with current issues; – updates obsolescent style elements. Removes Space Packet and renames the document.

NOTE – Substantive changes from the previous issue are indicated with change bars in the inside margin.

CONTENTS

CONTENTS (continued)

Figure Page

Table

1 INTRODUCTION

1.1 PURPOSE

The purpose of this Recommended Standard is to specify the Encapsulation Packet Protocol (EPP). This protocol is used to encapsulate higher layer protocol data units that are recognized by CCSDS over applicable ground-to-space or space-to-ground or space-to-space communications links using Space Data Link Protocols (references [1]-[5]) .

1.2 SCOPE

This Recommended Standard defines the Encapsulation Packet Protocol in terms of:

a) the service primitives provided to the users of this protocol;

b) the protocol data units employed by the service provider; and

c) the procedures performed by the service provider.

It does not specify:

a) individual implementations or products;

b) the implementation of service interfaces within real systems;

c) the methods or technologies required to perform the procedures; or

d) the management activities required to configure and control the service.

1.3 APPLICABILITY

This Recommended Standard applies to the creation of Agency standards and to the future data communications over space links between CCSDS Agencies in cross-support situations. The Recommended Standard includes comprehensive specification of the service for inter-Agency cross support. It is neither a specification of, nor a design for, real systems that may be implemented for existing or future missions.

The Recommended Standard specified in this document is to be invoked through the normal standards programs of each CCSDS Agency, and is applicable to those missions for which cross support based on capabilities described in this Recommended Standard is anticipated. Where mandatory capabilities are clearly indicated in sections of the Recommended Standard, they must be implemented when this document is used as a basis for cross support. Where options are allowed or implied, implementation of these options is subject to specific bilateral cross support agreements between the Agencies involved.

1.4 RATIONALE

The CCSDS Space Data Link protocols were primarily defined to carry application layer data that could be directly inserted into the data field of the protocol. In order to allow these space data link protocols to carry other protocol data units, such as internetworking protocols, or file transfer or message transfer protocols, a "shim" protocol was devised to specify unambiguously how this is to be performed.

1.5 DOCUMENT STRUCTURE

This document is divided into five numbered sections and three annexes:

a) Section 1 presents the purpose, scope, applicability and rationale of this Recommended Standard and lists the conventions, definitions, and references used throughout the document;

b) Section 2 provides an overview of the Encapsulation Packet Protocol;

c) Section 3 defines the service primitives provided for this protocol;

d) Section 4 specifies the protocol data units and procedures employed by the service provider;

e) Section 5 lists the managed parameters associated with this protocol;

f) Section 6 is discussion of security considerations pertinent to the specification;

g) Annex A lists all acronyms used within this document;

h) Annex B provides a list of informative references;

1.6 CONVENTIONS AND DEFINITIONS

1.6.1 DEFINITIONS

1.6.1.1 Definitions from the Open Systems Interconnection (OSI) Basic Reference Model

This Recommended Standard makes use of a number of terms defined in reference [6]. The use of those terms in this Recommended Standard shall be understood in a generic sense; i.e., in the sense that those terms are generally applicable to any of a variety of technologies that provide for the exchange of information between real systems. Those terms are:

a) blocking;

b) connection;

c) entity;

d) flow control;

e) protocol data unit;

f) real system;

g) segmenting;

h) service;

i) Service Access Point (SAP);

j) SAP address;

k) service data unit.

1.6.1.2 Definitions from OSI Service Definition Conventions

This Recommended Standard makes use of a number of terms defined in reference [77]. The use of those terms in this Recommended Standard shall be understood in a generic sense; i.e., in the sense that those terms are generally applicable to any of a variety of technologies that provide for the exchange of information between real systems. Those terms are:

a) indication;

b) primitive;

c) request;

d) service provider;

e) service user.

1.6.1.3 Terms Defined in This Recommended Standard

For the purposes of this Recommended Standard, the following definitions also apply. Many other terms that pertain to specific items are defined in the appropriate sections.

asynchronous: not *synchronous* (see below).

synchronous: of or pertaining to a sequence of events occurring in a fixed time relationship (within specified tolerance) to another sequence of events.

Encapsulation Idle Packet: an Encapsulation Packet that contains idle data.

Physical Channel: a stream of bits transferred over a space link in a single direction.

space link: a communications link between a spacecraft and its associated ground system, or between two spacecraft. A space link consists of one or more Physical Channels in one or both directions.

Commented [PS1]: **Are all of these really defined in this standard, or are they defined elsewhere and used here? ** now adjusted see below ****

Commented [GPC2R1]: Most of them are candidate for the SLS Glossary Magenta Book.
For some other terms we have discussions ongoing in SLS.
My suggestion is
-Reject all deletions for the time being and keep this section under review for the correct allocation.
-List proposed additions and check allocation and definition according to SLS discussions.

1.6.1.4 Terms Defined in CCSDS 133.0-B-2

delimited: having a known (and finite) length; applies to data in the context of data handling.

idle data: a fixed-length project specified 'idle' pattern of binary digits, whose assignment is a project design choice.

Commented [GPC3]: It may be it is a randomly generated sequence and not a defined pattern.

1.6.2 NOMENCLATURE

The following conventions apply throughout this Recommended Standard:

a) the words 'shall' and 'must' imply a binding and verifiable specification;

b) the word 'should' implies an optional, but desirable, specification;

c) the word 'may' implies an optional specification;

d) the words 'is', 'are', and 'will' imply statements of fact.

1.6.3 CONVENTIONS

In this document, the following convention is used to identify each bit in an N-bit field. The first bit in the field to be transmitted (i.e., the most left-justified when drawing a figure) is defined to be 'Bit 0'; the following bit is defined to be 'Bit 1' and so on up to 'Bit N-1'. When the field is used to express a binary value (such as a counter), the Most Significant Bit (MSB) shall be the first transmitted bit of the field, i.e., 'Bit 0' (see figure 1-1).

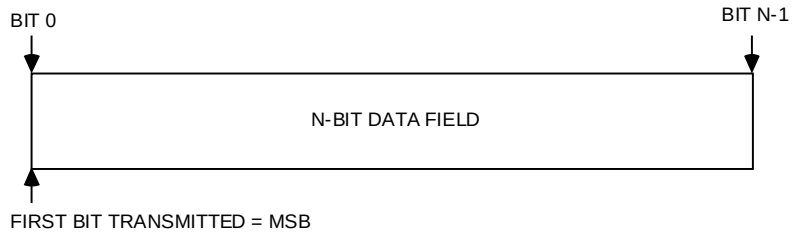


Figure 1-1: Bit Numbering Convention

In accordance with standard data-communications practice, data fields are often grouped into 8-bit 'words' which conform to the above convention. Throughout this Recommended Standard, such an 8-bit word is called an 'octet'.

The numbering for octets within a data structure starts with 0.

By CCSDS convention, all 'spare' bits shall be permanently set to value 'zero'.

1.7 REFERENCES

The following documents contain provisions which, through reference in this text, constitute provisions of this Recommended Standard. At the time of publication, the editions indicated were valid. All documents are subject to revision, and users of this Recommended Standard are encouraged to investigate the possibility of applying the most recent editions of the documents indicated below. The CCSDS Secretariat maintains a register of currently valid CCSDS Recommended Standards.

[1] *TM Space Data Link Protocol*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 132.0-B-1. Washington, D.C.: CCSDS, September 2003.

[2] *TC Space Data Link Protocol*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 232.0-B-2. Washington, D.C.: CCSDS, September 2010.

[3] *AOS Space Data Link Protocol*. Recommendation for Space Data System Standards, CCSDS 732.0-B-2. Blue Book. Issue 2. Washington, D.C.: CCSDS, July 2006.

[4] *Proximity-1 Space Link Protocol—Data Link Layer*. Issue 5. Recommendation for Space Data System Standards (Blue Book), CCSDS 211.0-B-5. Washington, D.C.: CCSDS, December 2013.

[5] [USLP Reference to be added: CCSDS 732.1-B-1].

[6] *Information Technology—Open Systems Interconnection—Basic Reference Model: The Basic Model*. 2nd ed. International Standard, ISO/IEC 7498-1:1994. Geneva: ISO, 1994.

[77] *Information Technology—Open Systems Interconnection—Basic Reference Model—Conventions for the Definition of OSI Services*. International Standard, ISO/IEC 10731:1994. Geneva: ISO, 1994.

[88] "Space Assigned Number Authority (SANA) Registry: Packet Version Number." Space Assigned Numbers Authority. Consultative Committee for Space Data Systems. http://sanaregistry.org/r/packet_version_number/.

[9] [SDLS Reference to be added].

[10] "Space Assigned Number Authority (SANA) Registry: Protocol Identifier for Encapsulation Service." Space Assigned Numbers Authority. Consultative Committee for Space Data Systems. http://sanaregistry.org/r/protocol_id/.

,[11] "Space Assigned Number Authority (SANA) Registry: Extended Protocol Identifiers." Space Assigned Numbers Authority. Consultative Committee for Space Data Systems. http://sanaregistry.org/r/extended_protocol_id/.

NOTE – Informative references are listed in annex B.

Cor. 2

Cor. 2

2 OVERVIEW

2.1 CONCEPT OF ENCAPSULATION PACKET PROTOCOL

The Encapsulation Packet Protocol (EPP) is used to transfer protocol data units recognized by CCSDS [10-11] that are *not directly transferred* by the Space Data Link Protocols (references [1]-[5]) over an applicable ground-to-space or space-to-ground or space-to-space communications links.

Data units that may be directly transferred by the Space Data Link Protocols have a Packet Version Number (PVN) authorized by CCSDS. (A list of the Packet Version Numbers presently authorized by CCSDS is contained in reference [8].) The main purpose of the Encapsulation Service is to provide a mechanism to transfer protocol data units *without an authorized PVN* over a space link.

The EPP is performed within the Data Link Layer of the OSI Basic Reference Model [6 (see figure 2-1). It utilizes the packet services of the Space Data Link Protocols defined in references [1]-[5], and therefore shall be used together with one of these references.

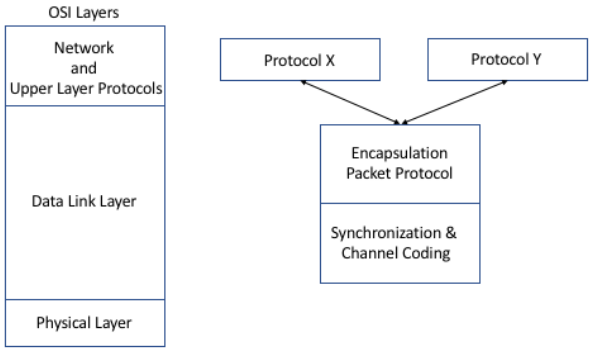


Figure 2-1: Concept of Encapsulation Packet Protocol

Figure 2-1 illustrates the concept of this protocol. Protocol data units of Protocols X and Y, which do not have an authorized PVN, are transferred with the EPP within the Data Link Layer. Protocol data units of Protocols X and Y are encapsulated in Encapsulation Packets defined in subsection 4.1 of this document, and are eventually transferred using one of the VC/MAP/Proximity-1 Packet Services of a Space Data Link Protocol. Management shall

Commented [GPC4]: Question for TOM: would be can usable in this sentence?

Commented [GPC5]: This statement shall be adequately discussed as it has very big implications on the structure we have defined till now.
PROPER DISCUSSION NEEDED AT SPRING 19 MEETING.
Of course I understand that this book already identified Encapsulation within the Data Link Layer as shown in Fig.2-1

Commented [GPC6]: If Tom can Edit the old figure to remove "Encapsulation Service, I would prefer keeping it.
IMO the new figure is not correct as does omit the protocol sublayer i.e. the Space Data Link Protocols.
Alternatively we may include a third "column" for CCSDS Protocols (as e.g. Figure 2-1 from https://public.ccsds.org/Pubs/132x0b2.pdf and other docs).

Commented [GPC7]: For the old figure (and for the new one) the question is whether EPP is really within the Data Link layer. If EPP is within that layer, than SPP is also there.
Note that EPP and SPP are both users of packet services provided by SDLProtocols.
FOR DISCUSSION

establish which Space Data Link Protocol is to be used to transfer encapsulated protocol data units.

Commented [GPC8]: FOR DISCUSSION: Should we use the plural; i.e. allowing an instance of the EPP to be able to access more than one SDLP? Should we reflect this in the managed parameters? ** Practically speaking, one SDLP is sufficient **

Commented [GPC9R8]: Some spacecract have more than one space link; e.g. ESA EO satellutes have a downlink with TM SDLP and one link with AOS SDLP. FOR DISCUSSION at Spring 2019 Meeting:

2.2 FEATURES OF ENCAPSULATION PACKET PROTOCOL

The EPP transfers a sequence of variable-length, delimited, octet-aligned protocol data units within the data field of a Space Data Link Protocol over a space link. A user of this protocol is a protocol entity that sends or receives protocol data units that do not have an authorized PVN.

A data unit supplied by the protocol user is encapsulated unchanged into an Encapsulation Packet and one and only one data unit is encapsulated into a single packet.

The protocol permits a data unit to be of any length which is an integral number of octets, and which is subject to the maximum and minimum sizes established by the project organization. Although the maximum length of a data unit that can be accommodated by an Encapsulation Packet is 4,294,967,287 octets, individual project organizations may establish the maximum and minimum sizes for the encapsulated data unit.

The point at which an instance of this protocol is provided to a user is called a Service Access Point (SAP) (reference [6]). Data units submitted to a SAP are processed in the order of submission. No processing order is maintained for data units submitted to different SAPs.

NOTE – Implementations may be required to perform flow control at an SAP between the service user and the service provider. However, CCSDS does not recommend a scheme for flow control between the user and the provider.

Features of the Encapsulation Packet Protocol are as follows:

a) Unidirectional (one way) service: one end of a connection can send, but not receive, data through the space link, while the other end can receive, but not send, data through the space link.

b) Asynchronous service: There are no timing relationships between the transfer of data units supplied by the user and any data transmission mechanism within the Data Link Layer. The user may request data transfer at any time, but there may be restrictions imposed by the service provider on the data generation rate.

c) Unconfirmed service: the sending user does not receive confirmation from the receiving end indicating that data has been received.

d) Incomplete service: the service does not guarantee completeness, but the service provider may signal gaps in the sequence of data units delivered to the receiving user.

e) Sequence preserving service: the sequence of data units supplied by the sending user is preserved through the transfer over the space link, although there may be gaps in the sequence of data units delivered to the receiving user.

2.3 ADDRESSING

A user of the EPP is identified by the Encapsulation Protocol Identifier (EPI).

The Encapsulation Packet Protocol Identifier is a Protocol ID defined in section 4 of this document.

Encapsulation Protocol Identifiers shall be registered as 'defined Protocol IDs' in reference [10].

Cor. 2

A SAP is identified by the combination of a PVN, an EPI, and a Space Data Link Protocol (SDLP) channel through which the data units supplied by the user are to be transferred.

2.4 PROTOCOL DESCRIPTION

The EPP is described in terms of:

a) the primitives provided to the users of this protocol;

b) the protocol data units employed by the protocol for encapsulation; and

c) the procedures performed by the protocol.

The primitives present an abstract model of the logical exchange of data and control information between the service provider and the service user. The definitions of primitives are independent of specific implementation approaches.

The protocol data unit (i.e., the Encapsulation Packet) defines the data structure in which data units supplied by the service user are encapsulated.

The procedure specifications define the procedures performed by the service provider for the transfer of data units. The definitions of procedures are independent of specific implementation methods or technologies.

3 SERVICE DEFINITION

3.1 OVERVIEW

This section provides service definition in the form of primitives, which present an abstract model of the logical exchange of data and control information between the service provider and the service user. The definitions of primitives are independent of specific implementation approaches.

The parameters of the primitives are specified in an abstract sense and specify the information to be made available to the user of the primitive. The way in which a specific implementation makes this information available is not constrained by this specification. In addition to the parameters specified in this section, an implementation may provide other parameters to the service user (e.g., parameters for controlling the service, monitoring performance, facilitating diagnosis, and so on).

3.2 PROTOCOL PARAMETERS

NOTE – The parameters used by the EPP primitives are described in subsections 3.2.1 through 3.2.5.

3.2.1 DATA UNIT

The parameter Data Unit is the service data unit transferred by the EPP, and it shall be a delimited, octet-aligned data unit.

Although the maximum length of a data unit that can be accommodated in an encapsulating packet is 4,294,967,287 octets for the Encapsulation Packet, individual project organizations may establish the maximum and minimum sizes for the encapsulated data unit.

3.2.2 SDLP_CHANNEL

3.2.2.1 The parameter SDLP_Channel is part of the SAP address of the EPP and it shall uniquely identify the channel of the underlying SDLP through which the Protocol Data Unit is to be transferred.

3.2.2.2 The contents of SDLP_Channel depend on the underlying SDLP service:

a) for the Virtual Channel Packet (VCP) service of TM (reference [1]), TC (reference [2]), or AOS (reference [3]), SDLP_Channel shall contain the Global Virtual Channel Identifier (GVCID);

b) for the MAP Packet (MAPP) service of TC (reference [2]) and USLP (reference [5]), SDLP_Channel shall contain the Global MAP ID (GMAP ID);

c) for the Packet Service of Proximity-1 (reference [4]), SDLP_Channel shall contain the Transfer Frame Version Number, the Spacecraft Identifier (SCID), the Physical Channel Identifier (PCID), the Data Field Construction ID (DFC_ID), and the Port ID.

NOTE – Definitions of Virtual Channel, MAP, GVCID, GMAP ID, Transfer Frame Version Number, SCID, PCID, and Port ID can be found in references [1]-[4].

3.2.3 EPI

The Encapsulation Protocol Identifier (EPI) is part of the SAP address of the EPP. It identifies the external protocol data unit to be encapsulated by this protocol. It is registered in SANA as the "Encapsulation Protocol Identifier" in reference [10] and defined in section 4 of this document.

3.3 PROTOCOL PRIMITIVES

3.3.1 GENERAL

The service primitives associated with this service are:

a) ENCAPSULATION.request. The ENCAPSULATION.request primitive shall be passed from the protocol user at the sending end to the service provider to request that a Protocol Data Unit be transferred, through the underlying Space Data Link Protocol, to the user at the receiving end.

b) ENCAPSULATION.indication. The ENCAPSULATION.indication shall be passed from the service provider to the protocol user at the receiving end in order to deliver a Protocol Data Unit.

3.3.2 ENCAPSULATION.REQUEST

3.3.2.1 Function

The ENCAPSULATION.request primitive shall be the service request primitive for the Encapsulation Packet Protocol.

3.3.2.2 Semantics

The ENCAPSULATION.request primitive shall provide parameters as follows:

ENCAPSULATION.request	(Protocol Data Unit, SDLP_Channel, EPI)

Cor. 2

NOTE – Information on the management of the SDLP_Channel and EPI parameters, and where applicable values are defined, is contained in 3.2.2, and reference [10] respectively.

3.3.2.3 When Generated

The ENCAPSULATION.request primitive shall be passed to the service provider to request it to send the Protocol Data Unit.

3.3.2.4 Effect On Receipt

Receipt of the ENCAPSULATION.request primitive shall cause the service provider to transfer the Protocol Data Unit.

3.3.2.5 Additional Comments

The ENCAPSULATION.request primitive shall be used to transfer Protocol Data Units across the space link through the underlying Space Data Link Protocol.

3.3.3 ENCAPSULATION.INDICATION

3.3.3.1 Function

The ENCAPSULATION.indication primitive shall be the service indication primitive for the Encapsulation Packet Protocol.

3.3.3.2 Semantics

The ENCAPSULATION.indication primitive shall provide parameters as follows:

ENCAPSULATION.indication	(Protocol Data Unit, SDLP_Channel, EPI)

NOTE – Information on the management of the SDLP_Channel and EPI parameter, and where applicable values are defined, is contained in 3.2.2, reference [8], and reference [10], respectively.

Cor. 2

3.3.3.3 When Generated

The ENCAPSULATION.indication primitive shall be passed from the service provider to the protocol user at the receiving end in order to deliver a Protocol Data Unit.

3.3.3.4 Effect On Receipt

The effect on receipt of the ENCAPSULATION.indication primitive by the protocol user is undefined.

3.3.3.5 Additional Comments

The ENCAPSULATION.indication primitive shall be used to deliver Protocol Data Units to the protocol user identified by the SDLP_Channel, and EPI.

4 DATA UNITS AND PROCEDURES

4.1 SECTION DELETED

4.2 ENCAPSULATION PACKET

4.2.1 GENERAL

An Encapsulation Packet shall encompass the major fields, positioned contiguously, in the following sequence:

a) Packet Header (1 to 8 octets, mandatory);

b) Encapsulated Protocol Data Unit (from 0 to 4,294,967,287 octets, optional).

An Encapsulation Packet shall consist of at least 1 and at most 4,294,967,295 octets.

NOTE – The maximum Encapsulation Packet length allowed by a particular spacecraft or ground implementation may be less than the maximum specified here.

The structural components of the Encapsulation Packet are shown in figure 4-1.

NOTE – The definition of the Encapsulation Packet Header, below, differs from and is incompatible with the initial definition contained in references [B2]-[B4]. The definition below replaces the initial definition and renders it obsolete.

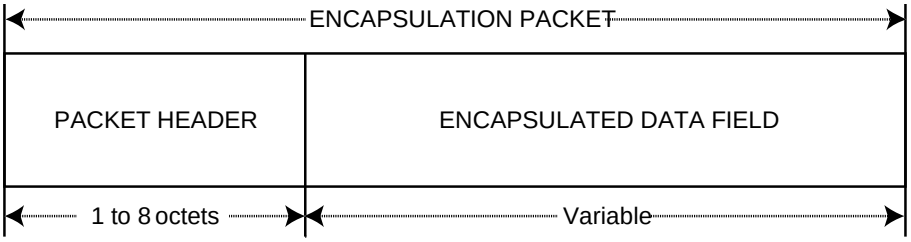


Figure 4-1: Encapsulation Packet Structural Components

4.2.2 ENCAPSULATION PACKET HEADER

4.2.2.1 General

4.2.2.1.1 The Packet Header is mandatory and it shall have a length of one, two, four, or eight octets. The Packet Header shall consist of three, four, six, or seven fields, positioned contiguously, in the following sequence:

a) Packet Version Number (3 bits, mandatory);

Commented [GPC10]: To avoid renumbering etc. it may be good keeping section 4.1 just to say "Space Packet is not allowed anymore" ** Fine to keep the same numbering but no need to mention SP – that kind of historical info belongs better in a GB and I am not proposing we write one **

Commented [GPC11R10]: Tom can solve this. ☺

Commented [GPC12]: For CCSDS Tech. Editor

b) Encapsulation Protocol ID (3 bits, mandatory);

c) Length of Length (2 bits, mandatory);

d) User Defined field (4 bits; mandatory in 4- and 8-octet headers; not used in 1- and 2-octet headers—see figure 4-2);

e) Encapsulation Protocol ID Extension field (4 bits; mandatory in 4- and 8-octet headers; not used in 1- and 2-octet headers—see figure 4-2);

f) CCSDS Defined field (2 octets; mandatory in 8-octet headers; not used in 1-, 2-, and 4-octet headers—see figure 4-2);

g) Packet Length (1, 2, or 4 octets; mandatory in 2-, 4-, and 8-octet headers; not used in 1-octet headers—see figure 4-2).

NOTE – The format of the Packet Header is shown in figure 4-2.

ENCAPSULATION PACKET HEADER

PACKET VERSION NUMBER 3 bits	PROTOCOL ID 3 bits	LENGTH OF LENGTH 2 bits	USER DEFINED FIELD 0 or 4 bits	PROTOCOL ID EXTENSION 0 or 4 bits	CCSDS DEFINED FIELD 0 or 2 octets	PACKET LENGTH 0 to 4 octets
'111'	'XXX'	'00'	0 bits	0 bits	0 octets	0 octets
'111'	'XXX'	'01'	0 bits	0 bits	0 octets	1 octet
'111'	'XXX'	'10'	4 bits	4 bits	0 octets	2 octets
'111'	'XXX'	'11'	4 bits	4 bits	2 octets	4 octets

Figure 4-2: Packet Header

Commented [GPC13]: Tom G. to replace "Protocol ID" with "Encapsulation Protocol ID" and "Protocol ID Extension" with "Encapsulation Protocol ID Extension"

4.2.2.1.2 An implementation of the Encapsulation Packet Protocol shall be capable of receiving Encapsulation Packets with fixed- or variable-length headers.

NOTE – An implementation on the transmitting end may choose to use either a fixed Packet Header size or adaptively/dynamically adjust the Packet Header size according to the payload size in order to optimize bandwidth use (minimize header overhead).

4.2.2.2 Packet Version Number

Bits 0-2 of the Packet Header shall contain the (Binary Encoded) Packet Version Number.

This 3-bit field shall identify the data unit as an Encapsulation Packet defined by this subsection; it shall be set to '111'.

NOTE – The Version Number is used to reserve the possibility of introducing other packet structures. This subsection defines 'Encapsulation Packet (Version 8 CCSDS Packet)' whose Binary Encoded Version Number is '111'.

4.2.2.3 Encapsulation Protocol ID (EPI)

Bits 3-5 of the Packet Header shall contain the Encapsulation Protocol ID.

The Encapsulation Protocol ID shall be used to identify the protocol whose data unit is encapsulated within the Encapsulation Packet.

Cor. 2

The Encapsulation Protocol IDs recognized by CCSDS for the Encapsulation Packet shall be registered in reference [10].

NOTES

1 The value '000' in the Protocol ID field signals that the packet is an Encapsulation Idle Packet. Encapsulation Idle Packets may be used to fill space in a fixed-length Transfer Frame used in references [1], [3], and [5].

2 The value '110' in the Encapsulation Protocol ID field signals that the 4-bit Encapsulation Protocol ID Extension field is used for protocol identification.

3 The value '111' in the Encapsulation Protocol ID field signals that the Encapsulated Data field contains mission-specific, privately defined data.

4.2.2.4 Length of Length

Bits 6-7 of the Packet Header shall contain the Length of Length field.

The Length of Length shall define the length of the Packet Length field.

The value of this field shall be interpreted as shown in table 4-1.

Table 4-1: Interpretation of Length of Length Field

Value of 'Length of Length' Field (binary)	Length of 'Packet Length' Field
00	0
01	1 octet
10	2 octets
11	4 octets

If the Length of Length field has the value '00' then the Protocol ID field shall have the value '000', indicating that the packet is an Encapsulation Idle Packet.

NOTE – If the Length of Length field has the value '00', then the Packet Length field and the Encapsulated Data Unit field are both absent from the packet. In this case, the length of the Encapsulation Packet is one octet.

4.2.2.5 User Defined Field

If present, the User Defined field shall be four bits in length and shall follow, without gap, the Length of Length field.

The User Defined field shall be used for sending mission-specific, privately defined header data.

4.2.2.6 Encapsulation Protocol ID Extension

If present, the Encapsulation Protocol ID Extension field shall be four bits in length and shall follow, without gap, the User Defined field.

If the Encapsulation Protocol ID field contains the value '110', then this field shall be used to identify the protocol whose data unit is encapsulated within the Encapsulation Packet. If the Encapsulation Protocol ID field does not contain the value '110', then this field is by convention set to 'all zeros'.

The extended Encapsulation Protocol IDs shall be registered in reference [11].

Cor. 2

4.2.2.7 CCSDS Defined Field

If present, the CCSDS Defined field shall be two octets in length and shall follow, without gap, the Encapsulation Protocol ID Extension field.

The CCSDS Defined field is reserved for future use by CCSDS and is by convention set to 'all zeros'.

4.2.2.8 Packet Length

If present, the Packet Length field shall be the final field in the Encapsulation Packet Header (see figure 4-2).

If the value of the Length of Length field is '00', the Packet Length field shall be absent. Otherwise, the Packet Length field shall contain a binary number corresponding to the total length of the Encapsulation Packet (in octets), including the Packet Header. See table 4-2.

NOTES

1 Although unlikely to be used in space, a 4-octet Packet Length field permits accommodating IPv6 (reference [B6]) 'Jumbograms' up to 4,294,967,287 ($=2^{32}-5$) octets in length.

2 If the Packet Length field is absent (i.e., the value of the Length of Length field is '00'), then the length of the Encapsulation Packet is one octet.

Table 4-2: Encapsulation Packet Lengths, Depending on the Length of the Packet Header

Length (*) of Packet Header	Number of octets in Packet Length Field	Minimum packet length	Maximum packet length	Minimum length of Encapsulated Data Field	Maximum length of Encapsulated Data Field (**)
1	Packet Length Field is absent	1	1	Encapsulated Data Field is absent	
2	1	2	255	0	253
4	2	4	65,535	0	65,531
8	4	8	4,294,967,295	0	4,294,967,287

(*) All lengths are given in octets.
(**) An implementation may establish a lower value for the maximum length of the Encapsulated Data field. The length of the Encapsulated Data Unit is the same as the length of the Encapsulated Data field.

4.2.3 ENCAPSULATED DATA FIELD

If present, the Encapsulated Data field shall follow, without gap, the Packet Length field. It shall consist of an integral number of octets.

The Encapsulated Data field shall contain the protocol data as indicated by the Encapsulation Protocol ID field, and by the Encapsulation Protocol ID Extension field if present.

If the Protocol ID field contains the value '000', then the Encapsulated Data field shall contain idle data.

Under the following conditions, the Encapsulated Data field shall be absent:

a) when the value of the Length of Length field is '00'; or

b) when the value of the Length of Length field is other than '00', and the packet length as indicated by the Packet Length field is equal to the length of the Encapsulation Packet Header.

If the Encapsulated Data field is absent then the Encapsulation Protocol ID field shall contain the value '000'.

NOTES

1 When the Encapsulation Packet is generated as a result of an ENCAPSULATION.request to the Encapsulation Packet Protocol, then the Encapsulated Data field contains the Protocol Data Unit supplied by the protocol user.

2 If the Encapsulation Protocol ID field contains the value '000', then the packet is an Encapsulation Idle Packet. Encapsulation Idle Packets may be used as Idle Packets by the TM Space Data Link Protocol (reference [1]) and by the AOS Space Data Link Protocol (reference [3]) and Unified Space Data Link Protocol (reference [5]). These protocols generate Idle Packets when needed to maintain synchronization of the data transport processes.

3 CCSDS does not specify the bit pattern of idle data.

4.3 PROCEDURES AT THE SENDING END

NOTE – This subsection describes procedures for providing the Encapsulation Packet Protocol at the sending end (see figure 4-3). The procedures described here are defined in an abstract sense and are not intended to imply any particular implementation approach of the service.

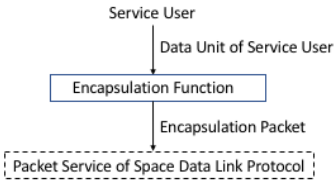


Figure 4-3: Internal Organization of Encapsulation Packet Protocol (Sending End)

The Encapsulation Function (see figure 4-3) shall be used to encapsulate data units supplied by the service users.

Commented [GPC14]: For consistency I think Figure should show:
- Protocol User
- Data Unit of Protocol User

Commented [GPC15]: Repaced with Encapsulation Packet Protocol for consistency with initial NOTE.

The Encapsulation Function receives data units from the service users. Any data unit that violates the limits of the size shall be rejected. Each valid protocol data unit shall be encapsulated, in an Encapsulation Packet, and passed to the Packet Service of the underlying Space Data Link Protocol.

Commented [GPC16]: Protocol user?

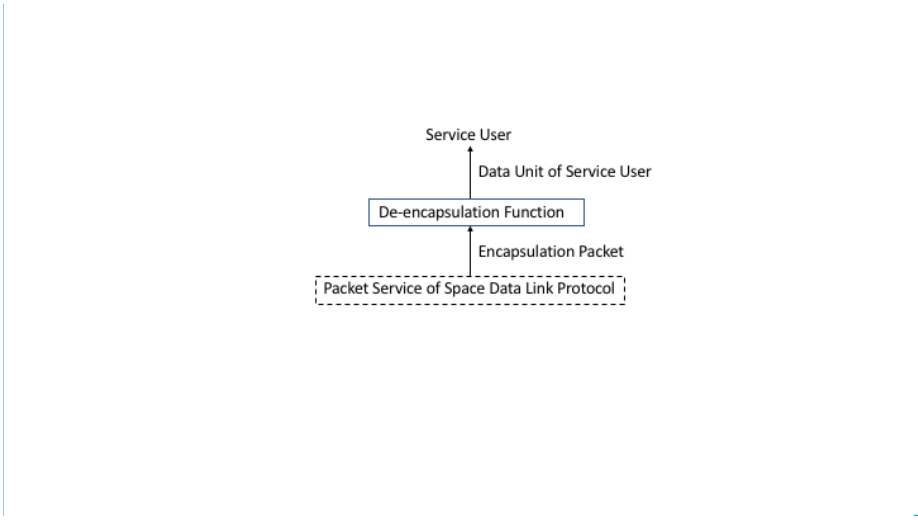
Commented [GPC17]: I would delete "protocol" as this is formally a SDU and not a PDU. Better keeping only data unit

Commented [GPC18]: For discussion: only one SDLP? ** yes, only one SDLP at a time **

Commented [GPC19R18]: For Discussion at Spring Meeting 2019

4.4 PROTOCOL PROCEDURES AT THE RECEIVING END

NOTE – This subsection describes procedures for providing the Encapsulation Packet Protocol at the receiving end (see figure 4-4). The procedures described here are defined in an abstract sense and are not intended to imply any particular implementation approach of the service.



Commented [GPC20]: For consistency I think Figure should show:
- Protocol User
- Data Unit of Protocol User

Figure 4-4: Internal Organization of Encapsulation Packet Protocol (Receiving End)

The De-encapsulation Function (see figure 4-4) shall be used to extract protocol data units supplied by the service users.

Commented [GPC21]: I would delete "protocol" as this is formally a SDU and not a PDU. Better keeping only data unit

The De-encapsulation Function shall receive Encapsulation Packets, from the Packet Service of the underlying Space Data Link Protocol, and extract original protocol data units by stripping the headers of the Encapsulation packets. Extracted protocol data units shall be delivered to the service user identified by the Encapsulation Protocol ID (EPI) or the extended EPI.

Commented [GPC22]: I would delete "protocol" as this is formally a SDU and not a PDU. Better keeping only data unit

5 MANAGED PARAMETERS

In order to conserve bandwidth on the space link, some parameters associated with the Encapsulation Packet Protocol (EPP) are handled by management, rather than by inline communications protocol. The managed parameters are those which tend to be static for long periods of time, and whose change generally signifies a major reconfiguration of the service provider associated with a particular mission. Through the use of a management system, management conveys the required information to the service provider.

The managed parameters are listed in table 5-1. These parameters are defined in an abstract sense, and are not intended to imply any particular implementation of a management system.

Table 5-1: Managed Parameters for Encapsulation Packet Protocol)

Managed Parameter	Allowed Values	Defined In
Minimum Data Unit Length (octets)	Integer	4.2
Maximum Data Unit Length (octets)	Integer	4.2
Valid Encapsulation Protocol Identifiers	Set of integers	Reference [10]
Valid Extended Encapsulation Protocol Identifiers	Set of integers	Reference [11]

Cor. 2

Cor. 2

6 SECURITY

Commented [GPC23]: I guess this chapter shall be removed as a new Annex for Security has been added. ** Tom Gannett to delete this Section **

ANNEX A

ACRONYMS

(INFORMATIVE)

This annex lists the acronyms used in this Recommended Standard.

AOS	Advanced Orbiting Systems
CCSDS	Consultative Committee for Space Data Systems
DFC_ID	Data Field Construction Identifier
EPP	Encapsulation Packet Protocol
EPI	Encapsulated Protocol Identifier
GVCID	Global Virtual Channel Identifier
IPv6	Internet Protocol, Version 6
OSI	Open Systems Interconnection
MSB	Most Significant Bit
PVN	Packet Version Number
SAP	Service Access Point
SDLP	Space Data Link Protocol
TFVN	Transfer Frame Version Number

ANNEX B

INFORMATIVE REFERENCES

(Informative)

Commented [MOU24]: Version numbers and dates throughout to be updated by Tom Gannett

[B1] *Organization and Processes for the Consultative Committee for Space Data Systems*. Issue 3. CCSDS Record (Yellow Book), CCSDS A02.1-Y-3. Washington, D.C.: CCSDS, July 2011.

[B2] *Advanced Orbiting Systems, Networks and Data Links: Architectural Specification*. Issue 3-S. Recommendation for Space Data System Standards (Historical Recommendation), CCSDS 701.0-B-3-S. Washington, D.C.: CCSDS, (June 2001) August 2005.

[B3] *Packet Telemetry*. Issue 5-S. Recommendation for Space Data System Standards (Historical Recommendation), CCSDS 102.0-B-5-S. Washington, D.C.: CCSDS, (November 2000) August 2005.

[B4] *Telecommand Part 3—Data Management Service*. Issue 2-S. Recommendation for Space Data System Standards (Historical Recommendation), CCSDS 203.0-B-2-S. Washington, D.C.: CCSDS, (June 2001) August 2005.

[B5] *Space Data Link Protocols—Summary of Concept and Rationale*. Issue 2. Report Concerning Space Data System Standards (Green Book), CCSDS 130.2-G-2. Washington, D.C.: CCSDS, November 2012.

[B6] S. Deering and R. Hinden. *Internet Protocol, Version 6 (IPv6) Specification*. RFC 1883. Reston, Virginia: ISOC, December 1995.

[B7] *The Application of CCSDS Protocols to Secure Systems*. Issue 2. Report Concerning Space Data System Standards (Green Book), CCSDS 350.0-G-2. Washington, D.C.: CCSDS, January 2006.

NOTE – Normative references are listed in subsection 1.7.

ANNEX C

SECURITY, SANA, AND PATENT CONSIDERATIONS

(INFORMATIVE)

Commented [GPC25]: CCSDS editor to fix section numbering.

A1 SECURITY CONSIDERATIONS

Security Considerations at the Data Link Layer are addressed by the *Space Data Link Security Protocol* (reference [SDLS Reference to be added]).

A2 SANA CONSIDERATIONS

A2.1 GENERAL

This section contains two modified SANA registry requests for the Encapsulation Packet Protocol.

A2.2 NEW REGISTRY REQUESTS

None

A2.3 MODIFIED REGISTRIES

A2.3.1 ~~Protocol_ID (old) changed~~ Keep SANA registry but Simply change the descriptive name in the SANA registry from "Protocol Identifer for Encapsulation Service" to "Protocol Identifer for Encapsulation Packet Protocol"

Reference: CCSDS 133.1-B-2

A2.3.2 ~~Extended_Protocol_ID (old) changed to Encapsulation Packet Protocol ID Extension~~ Keep SANA registry but Simply change the descriptive name in the SANA registry from "Extended Protocol Identifer for Encapsulation Service" to "Extended Protocol Identifer for Encapsulation Packet Protocol" **

Reference: CCSDS 133.1-B-2

Commented [GPC26]: Why renaming?
The full name shown at https://sanaregistry.org/r/extended_protocol_id/ is "Extended Protocol Identifier for Encapsulation Service". I do not see a need for this change (except the change from Service to Packet) **
Agreed. We do not have to rename the registry. We only need to rename the description of the registry **

NOTE FOR READERS: The older Annex C containing the term changed with respect to document [B2]-[B4] has been deleted.