

**Draft Recommendation for
Space Data System Standards**

**UNIFIED SPACE
DATA LINK
PROTOCOL**

DRAFT RECOMMENDED STANDARD

CCSDS 732.1-R-3

RED BOOK
August 2017



CCSDS

The Consultative Committee for Space Data Systems

**Draft Recommendation for
Space Data System Standards**

**UNIFIED SPACE
DATA LINK
PROTOCOL**

DRAFT RECOMMENDED STANDARD

CCSDS 732.1-R-3

RED BOOK

August 2017

AUTHORITY

Issue:	Red Book, Issue 3
Date:	August 2017
Location:	Not Applicable

(WHEN THIS RECOMMENDED STANDARD IS FINALIZED, IT WILL CONTAIN THE FOLLOWING STATEMENT OF AUTHORITY:)

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 *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4), and the record of Agency participation in the authorization of this document can be obtained from the CCSDS Secretariat at the e-mail address below.

This document is published and maintained by:

CCSDS Secretariat
National Aeronautics and Space Administration
Washington, DC, USA
E-mail: secretariat@mailman.ccsds.org

STATEMENT OF INTENT

(WHEN THIS RECOMMENDED STANDARD IS FINALIZED, IT WILL CONTAIN THE FOLLOWING 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:

- o 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.
- o 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.
- o 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 technical 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 Unified Space Data Link Protocol described herein is intended for missions that are cross-supported between Agencies of the CCSDS.

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Recommended Standard is therefore subject to CCSDS document management and change control procedures, which are defined in the *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4). 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 sent to the CCSDS Secretariat at the e-mail 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.
- 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 (DLR)/Germany.
- European Space Agency (ESA)/Europe.
- Federal Space Agency (FSA)/Russian Federation.
- Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
- Japan Aerospace Exploration Agency (JAXA)/Japan.
- National Aeronautics and Space Administration (NASA)/USA.
- UK Space Agency/United Kingdom.

Observer Agencies

- Austrian Space Agency (ASA)/Austria.
- Belgian Federal Science Policy Office (BFSP0)/Belgium.
- Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
- China Satellite Launch and Tracking Control General, Beijing Institute of Tracking and Telecommunications Technology (CLTC/BITTT)/China.
- Chinese Academy of Sciences (CAS)/China.
- Chinese Academy of Space Technology (CAST)/China.
- Commonwealth Scientific and Industrial Research Organization (CSIRO)/Australia.
- Danish National Space Center (DNSC)/Denmark.
- Departamento de Ciência e Tecnologia Aeroespacial (DCTA)/Brazil.
- Electronics and Telecommunications Research Institute (ETRI)/Korea.
- 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.
- Korea Aerospace Research Institute (KARI)/Korea.
- Ministry of Communications (MOC)/Israel.
- Mohammed Bin Rashid Space Centre (MBRSC)/United Arab Emirates.
- National Institute of Information and Communications Technology (NICT)/Japan.
- National Oceanic and Atmospheric Administration (NOAA)/USA.
- National Space Agency of the Republic of Kazakhstan (NSARK)/Kazakhstan.
- National Space Organization (NSPO)/Chinese Taipei.
- Naval Center for Space Technology (NCST)/USA.
- Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- Scientific and Technological Research Council of Turkey (TUBITAK)/Turkey.
- South African National Space Agency (SANSA)/Republic of South Africa.
- Space and Upper Atmosphere Research Commission (SUPARCO)/Pakistan.
- Swedish Space Corporation (SSC)/Sweden.
- Swiss Space Office (SSO)/Switzerland.
- United States Geological Survey (USGS)/USA.

PREFACE

This document is a draft CCSDS Recommended Standard. Its 'Red Book' status indicates that the CCSDS believes the document to be technically mature and has released it for formal review by appropriate technical organizations. As such, its technical contents are not stable, and several iterations of it may occur in response to comments received during the review process.

Implementers are cautioned **not** to fabricate any final equipment in accordance with this document's technical content.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

DOCUMENT CONTROL

Document	Title	Date	Status
CCSDS 732.1-R-1	Unified Space Data Link Protocol, Draft Recommended Standard, Issue	August 2016	Original draft issue (superseded)
CCSDS 732.1-R-2	Unified Space Data Link Protocol, Draft Recommended Standard, Issue 2	March 2017	Draft issue 2 (superseded)
CCSDS 732.1-R-3	Unified Space Data Link Protocol, Draft Recommended Standard, Issue 3	August 2017	Current draft: – incorporates changes resulting from review of draft issue 2 (see notes).

NOTES

- 1 For readability, markup is limited to change bars in the right margin indicating where substantive changes from draft issue 2 have been introduced.
- 2 The content of annex A (formerly to be supplied) is new in its entirety; it is not marked with change bars.

CONTENTS

<u>Section</u>	<u>Page</u>
1 INTRODUCTION.....	1-1
1.1 PURPOSE.....	1-1
1.2 SCOPE.....	1-1
1.3 APPLICABILITY.....	1-1
1.4 RATIONALE.....	1-2
1.5 DOCUMENT STRUCTURE.....	1-2
1.6 CONVENTIONS AND DEFINITIONS.....	1-3
1.7 REFERENCES.....	1-7
2 OVERVIEW.....	2-1
2.1 CONCEPT OF UNIFIED SPACE DATA LINK PROTOCOL.....	2-1
2.2 OVERVIEW OF SERVICES.....	2-6
2.3 OVERVIEW OF FUNCTIONS.....	2-15
2.4 SERVICES ASSUMED FROM LOWER LAYERS.....	2-18
3 SERVICE DEFINITION.....	3-1
3.1 OVERVIEW.....	3-1
3.2 SOURCE DATA.....	3-1
3.3 MAP PACKET SERVICE.....	3-3
3.4 MAP ACCESS SERVICE.....	3-9
3.5 MAP OCTET STREAM SERVICE.....	3-14
3.6 MASTER CHANNEL OPERATIONAL CONTROL FIELD SERVICE.....	3-18
3.7 VIRTUAL CHANNEL FRAME SERVICE.....	3-22
3.8 MASTER CHANNEL FRAME SERVICE.....	3-25
3.9 INSERT SERVICE.....	3-28
3.10 COPS MANAGEMENT SERVICE.....	3-31
4 PROTOCOL SPECIFICATION WITHOUT SDLS OPTION.....	4-1
4.1 PROTOCOL DATA UNIT.....	4-1
4.2 PROTOCOL PROCEDURES AT THE SENDING END.....	4-21
4.3 PROTOCOL PROCEDURES AT THE RECEIVING END.....	4-36
5 MANAGED PARAMETERS WITHOUT SDLS OPTION.....	5-1
5.1 OVERVIEW.....	5-1
5.2 MANAGED PARAMETERS FOR A PHYSICAL CHANNEL.....	5-2
5.3 MANAGED PARAMETERS FOR A MASTER CHANNEL.....	5-3

CONTENTS (continued)

<u>Section</u>	<u>Page</u>
5.4 MANAGED PARAMETERS FOR A VIRTUAL CHANNEL	5-4
5.5 MANAGED PARAMETERS FOR A MAP CHANNEL	5-6
5.6 MANAGED PARAMETERS FOR PACKET TRANSFER.....	5-6
6 PROTOCOL SPECIFICATION WITH SDLS OPTION.....	6-1
6.1 OVERVIEW	6-1
6.2 USE OF SDLS PROTOCOL.....	6-1
6.3 USLP TRANSFER FRAME WITH SDLS	6-1
6.4 SENDING END PROTOCOL PROCEDURES WITH SDLS.....	6-4
6.5 RECEIVING END PROTOCOL PROCEDURES WITH SDLS.....	6-7
6.6 MANAGED PARAMETERS WITH SDLS	6-9
ANNEX A PROTOCOL IMPLEMENTATION CONFORMANCE STATEMENT (PICS) PROFORMA (NORMATIVE).....	A-1
ANNEX B FRAME ERROR CONTROL FIELD CODING PROCEDURES (NORMATIVE)	B-1
ANNEX C RELATIONSHIP OF PROXIMITY-1 (VERSION-3) TO USLP (VERSION-4) TRANSFER FRAMES (NORMATIVE).....	C-1
ANNEX D SECURITY, SANA, AND PATENT CONSIDERATIONS (INFORMATIVE)	D-1
ANNEX E INFORMATIVE REFERENCES (INFORMATIVE)	E-1
ANNEX F ABBREVIATIONS AND ACRONYMS (INFORMATIVE).....	F-1

Figure

1-1 Bit Numbering Convention.....	1-6
2-1 Relationship with OSI Layers.....	2-1
2-2 Relationships between Channels	2-5
2-3 Asynchronous Service Model.....	2-9
2-4 Synchronous Service Model	2-10
2-5 Internal Organization of Protocol Entity (Sending End).....	2-16
2-6 Internal Organization of Protocol Entity (Receiving End).....	2-16
2-7 Unified Space Data Link Protocol Channel Tree	2-17
4-1 USLP Transfer Frame Structural Components.....	4-1
4-2 Non-truncated Transfer Frame Primary Header	4-2
4-3 Truncated Transfer Frame Primary Header	4-5
4-4 Transfer Frame Data Field.....	4-11
4-5 Transfer Frame Data Field Header	4-12
4-6 Internal Organization of Protocol Entity (Sending End).....	4-21

CONTENTS (continued)

<u>Figure</u>	<u>Page</u>
4-7 Abstract Model of Packet Processing Function for Fixed TFDFs.....	4-23
4-8 Abstract Model of MAP Packet Processing Function for a Variable-Length TFDF .	4-24
4-9 Abstract Model of MAPA_SDU Generation Function for Fixed-Length TFDF	4-26
4-10 Abstract Model of MAPA_SDU Generation Function for Variable-Length TFDF...	4-27
4-11 Abstract Model of the MAP Octet Stream Processing Function.....	4-28
4-12 Abstract Model of MAP Multiplexing Function.....	4-29
4-13 Abstract Model of Virtual Channel Generation Function.....	4-31
4-14 Abstract Model of Virtual Channel Multiplexing Function.....	4-32
4-15 Abstract Model of Master Channel Multiplexing Function.....	4-33
4-16 Abstract Model of All Frames Generation Function.....	4-35
4-17 Internal Organization of Protocol Entity (Receiving End).....	4-36
4-18 Abstract Model of MAP Packet Extraction Function for Fixed-Length TFDFs.....	4-38
4-19 Abstract Model of MAP Packet Extraction Function for Variable-Length TFDFs ...	4-39
4-20 Abstract Model of MAPA_SDU Extraction Function for Fixed-Length TFDFs.....	4-41
4-21 Abstract Model of MAPA_SDU Extraction Function for Variable-Length TFDFs ..	4-42
4-22 Abstract Model of MAP Octet Stream Extraction Function.....	4-43
4-23 Abstract Model of MAP Demultiplexing Function.....	4-44
4-24 Abstract Model of Virtual Channel Reception Function.....	4-45
4-25 Abstract Model of Virtual Channel Demultiplexing Function.....	4-46
4-26 Abstract Model of Master Channel Demultiplexing Function.....	4-47
4-27 Abstract Model of All Frames Reception Function.....	4-49
6-1 Frame without SDLS Compared to Frame with SDLS	6-2
B-1 Logic Diagram of the CRC-16 Encoder.....	B-2
B-2 Logic Diagram of the CRC-16 Decoder.....	B-3
B-3 A Possible Implementation of the CRC-32 Encoder.....	B-4
B-4 A Possible Implementation of the CRC-32 Decoder.....	B-5
C-1 Transfer Frame Data Field Header for Proximity Operations.....	C-4

Table

2-1 Summary of Services Provided by Unified Space Data Link Protocol.....	2-11
4-1 Interpretation of the Bypass/Sequence Control and Protocol Control Command Flags.....	4-7
4-2 Interpretation of the Virtual Channel Frame Count Length.....	4-8
4-3 Summary of the TFDZ Construction Rules.....	4-15
5-1 Managed Parameters for a Physical Channel.....	5-2
5-2 Managed Parameters for a Master Channel.....	5-3
5-3 Managed Parameters for a Virtual Channel.....	5-4
5-4 Managed Parameters for a MAP Channel.....	5-6

CONTENTS (continued)

<u>Table</u>	<u>Page</u>
5-5 Managed Parameters for Packet Transfer.....	5-6
6-1 Additional Managed Parameters for a Virtual Channel when the Unified Space Data Link Protocol Supports SDLS.....	6-9
A-1 USLP Service Data Units	A-4
A-2 Service Parameters.....	A-4
A-3 Service Primitives.....	A-6
A-4 USLP Protocol Data Unit	A-6
A-5 Protocol Procedures.....	A-7
A-6 Management Parameters.....	A-8
A-7 Protocol Specification with SDLS Option.....	A-10
A-8 Additional Managed Parameters with SDLS Option.....	A-10
A-9 Frame Error Control Field Coding Procedures.....	A-10
A-10 Relationship of Version-3 and Version-4 Transfer Frames.....	A-10
C-1 Relationship of Transfer Frame Header Fields between Version-3 and Version-4 Frames.....	C-1
C-2 TFDZ Construction Rules.....	C-2

1 INTRODUCTION

1.1 PURPOSE

The purpose of this Recommended Standard is to specify the Unified Space Data Link Protocol (USLP). This protocol is a Data Link Layer protocol (see reference [1]) to be used over space-to-ground, ground-to-space, or space-to-space communications links by space missions.

1.2 SCOPE

This Recommended Standard defines the USLP in terms of:

- a) the services provided to the users of this protocol;
- b) the protocol data units employed by the protocol; and
- c) the procedures performed by the protocol.

It does not specify:

- a) individual implementations or products;
- b) the implementation of service interfaces within real systems;
- c) the protocol procedures specified in both the COP-1 (reference [9]) and the COP-P (reference [10]);
- d) the security services specified in the SDLS protocol (reference [14]);
- e) the flow control;
- f) the methods or technologies required to perform the procedures; or
- g) the management activities required to configure and control the protocol.

1.3 APPLICABILITY

This Recommended Standard applies to the creation of Consultative Committee for Space Data Systems (CCSDS) Agency standards and to future data communications over space links between CCSDS Agencies in cross-support situations. The Recommended Standard includes comprehensive specification of the services and protocol 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 believes it is important to document the rationale underlying the recommendations chosen, so that future evaluations of proposed changes or improvements will not lose sight of previous decisions. The USLP Green Book (reference [E11]) contains further details including the applicability to various space links and data rates.

1.5 DOCUMENT STRUCTURE

This document is divided into six numbered sections and six 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 Recommended Standard;
- b) section 2 provides an overview of USLP;
- c) section 3 defines the services provided by the protocol entity;
- d) section 4 specifies the protocol data units and procedures employed by the protocol entity;
- e) section 5 specifies the managed parameters used by the protocol entity;
- f) section 6 specifies the protocol entity with support for the Space Data Link Security Protocol;
- g) annex A provides the Protocol Implementation Conformance Statement (PICS) proforma;
- h) annex B specifies Frame Error Control Field procedures;
- i) annex C specifies the relationship of Proximity-1 (Version-3) to USLP (version-4) Transfer Frames;
- j) annex D provides the security, Space Assigned Numbers Authority (SANA), and patent considerations;
- k) annex E provides a list of informative references;
- l) annex F lists all acronyms used within this document.

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 [1]. The use of those terms in this Recommended Standard is to 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) Data Link Layer;
- d) entity;
- e) flow control;
- f) Network Layer;
- g) peer entities;
- h) Physical Layer;
- i) protocol control information;
- j) protocol data unit;
- k) real system;
- l) segmenting;
- m) service;
- n) service access point, SAP;
- o) SAP address;
- p) 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 [2]. The use of those terms in this Recommended Standard is to 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) confirmation;

- b) indication;
- c) primitive;
- d) request;
- e) response;
- f) service provider;
- g) service user.

1.6.1.3 Terms Defined in *Proximity-1 Space Link Protocol—Data Link Layer*

This Recommended Standard makes use of the following terms defined in reference [10]:

caller and responder: Initiator and receiver, respectively, in a Proximity space link session. A **caller transceiver** is the initiator of the link establishment process and manager of negotiation (if required) of the session. A **responder transceiver** typically receives link establishment parameters from the caller. The caller initiates communication between itself and a responder on a prearranged communications channel with predefined controlling parameters. As necessary, the caller and responder may negotiate the controlling parameters for the session (at some level between fully controlled and completely adaptive).

forward link: That portion of a space link in which the caller transmits and the responder receives (typically a command link).

Port ID: Identifier of the logical or physical port that is the destination for a user's service data unit.

return link: That portion of a space link in which the responder transmits and the caller receives (typically a telemetry link).

1.6.1.4 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.

aperiodic: Not *periodic* (see **periodic**).

asynchronous: Not *synchronous* (see **synchronous**).

commutation: A function of concatenating, according to the formatting rule specified by the protocol definition, multiple data units, each from a different service, in a single protocol data unit sharing the same identifier.

decommutation: A function of decapsulating multiple data units, according to the formatting rule specified by the protocol definition, into separate data units, each from a different service.

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

Isochronous: Characterized by occurring at equal intervals of time.

Mission Phase: A period of a mission during which specified communications characteristics are fixed. The transition between two consecutive Mission Phases may cause an interruption of the communications services.

periodic: Of or pertaining to a sequence of events in which each event occurs at a fixed time interval (within specified tolerance) after the previous event in the sequence.

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.

synchronous: Of or pertaining to a sequence of events occurring in a fixed time relationship (within specified tolerance) to another sequence of events. It should be noted that 'synchronous' does not necessarily imply 'periodic' or 'constant rate'.

USLP Transfer Frame: The protocol data unit of the Unified Space Data Link Protocol.

NOTE – Within this document, depending on context, any of the terms 'Transfer Frame', 'Frame', and 'Version-4 (Transfer) Frame' may be used to refer to the 'USLP Transfer Frame'.

1.6.1.5 Terms Adapted from *Communications Operation Procedure-1* and *Proximity-1 Space Link Protocol—Data Link Layer*

The term **Frame Operation Procedure (FOP)** is used in this Recommended Standard to refer to both FOP-1 (see reference [9]) and FOP-P (see reference [10]). Similarly, the term **Frame Acceptance and Reporting Mechanism (FARM)** is used in this Recommended Standard to refer to both FARM-1 (see reference [9]) and FARM-P (see reference [10]). In both cases, interpretation of which term (-1 or P) applies depends on which Communications Operation Procedure, COP-1 (see reference [9]) or COP-P (see reference [10]), governs the context.

1.6.2 NOMENCLATURE

1.6.2.1 Normative Text

The following conventions apply for the normative specifications in 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.

NOTE – These conventions do not imply constraints on diction in text that is clearly informative in nature.

1.6.2.2 Informative Text

In the normative sections of this document, informative text is set off from the normative specifications either in notes or under one of the following subsection headings:

- Overview;
- Background;
- Rationale;
- Discussion.

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).

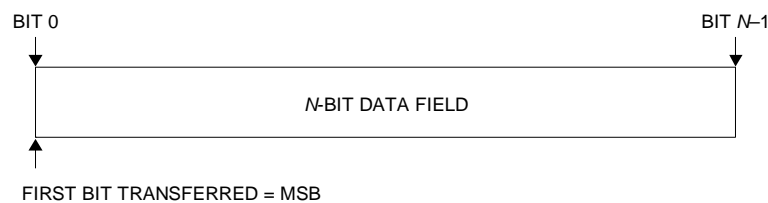


Figure 1-1: Bit Numbering Convention

In accordance with standard data-communications practice, data fields are often grouped into eight-bit ‘words’ which conform to the above convention. Throughout this Recommended Standard, such an eight-bit word is called an ‘octet’. The numbering for octets within a data structure starts with zero. By CCSDS convention, all ‘spare’ bits shall be permanently set to ‘0’.

1.7 REFERENCES

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

- [1] *Information Technology—Open Systems Interconnection—Basic Reference Model: The Basic Model*. 2nd ed. International Standard, ISO/IEC 7498-1:1994. Geneva: ISO, 1994.
- [2] *Information Technology—Open Systems Interconnection—Basic Reference Model—Conventions for the Definition of OSI Services*. International Standard, ISO/IEC 10731:1994. Geneva: ISO, 1994.
- [3] *TM Synchronization and Channel Coding*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 131.0-B-2. Washington, D.C.: CCSDS, August 2011.
- [4] *Flexible Advanced Coding and Modulation Scheme for High Rate Telemetry Applications*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 131.2-B-1. Washington, D.C.: CCSDS, March 2012.
- [5] *CCSDS Space Link Protocols over ETSI DVB-S2 Standard*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 131.3-B-1. Washington, D.C.: CCSDS, March 2013.
- [6] *TC Synchronization and Channel Coding*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 231.0-B-2. Washington, D.C.: CCSDS, September 2010.
- [7] *Proximity-1 Space Link Protocol—Coding and Synchronization Sublayer*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 211.2-B-2. Washington, D.C.: CCSDS, December 2013.
- [8] “Registries.” Space Assigned Numbers Authority. <http://sanaregistry.org/r/>.
- [9] *Communications Operation Procedure-1*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 232.1-B-2. Washington, D.C.: CCSDS, September 2010.

- [10] *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.
- [11] *CCSDS Global Spacecraft Identification Field Code Assignment Control Procedures*. Issue 6. Recommendation for Space Data System Standards (Blue Book), CCSDS 320.0-B-6. Washington, D.C.: CCSDS, October 2013.
- [12] *Space Packet Protocol*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 133.0-B-1. Washington, D.C.: CCSDS, September 2003.
- [13] *Encapsulation Service*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 133.1-B-2. Washington, D.C.: CCSDS, October 2009.
- [14] *Space Data Link Security Protocol*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 355.0-B-1. Washington, D.C.: CCSDS, September 2015.
- [15] *Space Data Link Security Protocol—Extended Procedures*. Recommendation for Space Data System Standards, CCSDS 355.1. Forthcoming.

NOTE – Informative references are listed in annex E.

2 OVERVIEW

2.1 CONCEPT OF UNIFIED SPACE DATA LINK PROTOCOL

2.1.1 ARCHITECTURE

The Unified Space Data Link Protocol is a Data Link Layer protocol (see reference [1]) to be used by space missions. This protocol has been designed to meet the requirements of space missions for efficient transfer of space application data of various types and characteristics over space-to-ground, ground-to-space, or space-to-space communications links (hereafter called space links).

Figure 2-1 illustrates the relationship of this protocol to the reference model of Open Systems Interconnection (reference [1]). Two sublayers of the Data Link Layer are defined for CCSDS space link protocols as shown in reference [E2]. USLP corresponds to the Data Link Protocol Sublayer and provides functions for transferring various data using a variable-length (or fixed-length as a subset) protocol data unit called the Transfer Frame. The Synchronization and Channel Coding Sublayer provides additional functions necessary for transferring Transfer Frames over a space link. These functions are: delimiting/synchronizing Transfer Frames, error-correction coding/decoding, and bit transition generation/removal (optional). Depending on the coding scheme applied some of the above-mentioned functions are optional or performed differently as explained later in this document. For the Synchronization and Channel Coding Sublayer, the set of Synchronization and Channel Coding Recommended Standards (references [3] through [7]) are to be used with USLP with constraints listed later in this document. How USLP is used in overall space data systems is shown in references [E2] through [E4] and [E11].

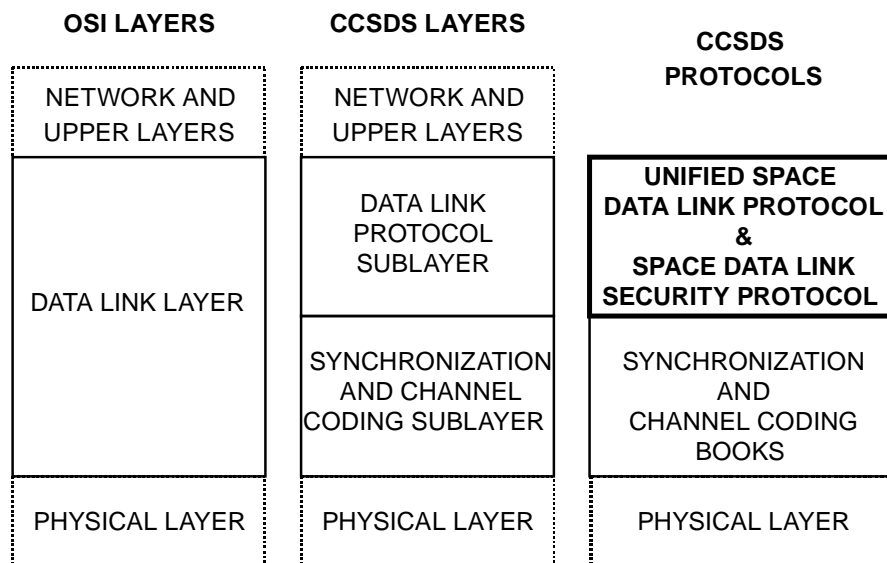


Figure 2-1: Relationship with OSI Layers

2.1.2 PROTOCOL FEATURES

2.1.2.1 Transfer Frames, Virtual Channels, and Multiplexer Access Points

USLP provides the users with several services to transfer service data units over a space link. These protocol data units used by USLP to provide the services are known as USLP Transfer Frames (unless otherwise stated, the terms ‘Transfer Frame’, ‘Frame’, and ‘Version-4 (Transfer) Frame’ in this document refer to the USLP Transfer Frame). Each Transfer Frame contains a primary header that provides protocol control information, identifying the length of the Frame and signaling the inclusion of selected fields. The Transfer Frames carry upper-layer service data units within the Transfer Frame Data Field (TFDF). The TFDF contains a header that identifies both how the data field is organized and the protocol to which the service data unit(s) are associated.

A key feature of space data link protocols, including USLP, is the concept of ‘Virtual Channels’ (VCs). The Virtual Channel facility allows one Physical Channel to be shared among multiple higher-layer data streams, each of which may have different service requirements. Of particular importance is the Quality of Service (QoS) attribute associated with a VC. A single Physical Channel may therefore be divided into several separate logical data channels, each known as a ‘Virtual Channel’. Each Transfer Frame transferred over a Physical Channel belongs to one of the VCs of the Physical Channel.

Moreover, this protocol enables service data units from different sources to be multiplexed together in one VC using ‘Multiplexer Access Points’ (MAPs). MAP_ID assignments allow service data units arriving at a SAP at the sending end to be transferred to a SAP with the corresponding MAP_ID at the receiving end. Each SAP can be associated with a specific protocol. USLP enables the transfer of CCSDS Packets, SDUs associated with protocols registered with SANA, or user defined data. Reference [8] is the SANA registry of CCSDS-recognized protocol IDs.

2.1.2.2 Additional USLP Features

USLP has a larger maximum Transfer Frame size than previous CCSDS space data link protocols in order to reduce the operational Frame handling process for high-rate missions. The protocol also has increased the capability for identifying (using a larger addressing space) more spacecraft than previous CCSDS space data link protocols. Moreover, it provides the means to transfer protocol data units from CCSDS-recognized protocols within the Frame via the CCSDS Encapsulation Service (reference [13]). USLP provides a configurable-sized sequence counter in the Frame Primary Header that is incremented to allow the receiving entity to determine if any Frames are missing.

2.1.2.3 Efficient Data Transfer

USLP provides the users with several services to transfer service data units over a space link. USLP performs (1) segmentation and blocking (i.e., aggregation) of service data units and (2) transmission control of service data units.

Because the underlying space link inherently includes a noisy signal path, there is a finite probability that it will introduce an error. It may be desirable to break large service data units into relatively small pieces when the frame error rate for large frames is higher than required by the mission. Under those conditions each piece has a lower probability of being invalidated by transmission error than if the entire service data unit were sent contiguously. System throughput efficiency may be improved because only small pieces have to be retransmitted when errors are detected. However, there may also be situations in which the service data units are very small. For efficient transfer of service data units, it is desirable to group these small units into larger pieces. USLP provides the capability to break large service data units into relatively small pieces (segmentation) and to group small service data units into larger pieces (blocking).

The Communications Operations Procedure (COP) as described in 2.1.2.4 below controls the transmission of service data units through the space link, performing retransmissions needed to ensure delivery of service data units in sequence and without gaps or duplication.

In addition, a systematic repeated frame retransmission mechanism for use on deep space links may optionally be provided by the Synchronization and Channel Coding Sublayer when reference [6] is applied (see 2.4.2).

2.1.2.4 Communications Operation Procedure (optional)

Within this document, the term COP refers to both the Communications Operation Procedure-1 (COP-1) (reference [9]), used in near-Earth and deep-space applications, and also the Communication Operations Procedure for Proximity links (COP-P) within the Proximity-1 Space Data Link Protocol (reference [10]). The Proximity-1 Green Book (reference [E14]) provides additional details.

The use of either the COP-1 or COP-P procedures is optional; both are compatible with USLP. The Protocol Data Units (CLCW for COP-1 and PLCW for COP-P) and operational procedures for COP-1 and COP-P are not identical but similar and are transparent to USLP. For this reason the generic terms Frame Acceptance and Reporting Mechanism (FARM) and Frame Operation Procedure (FOP) are used.

The controlling specifications for the logical operations which must be executed to perform both the COP-1 and COP-P are contained in more detailed CCSDS Recommended Standards (references [9] and [10]). In the event of any conflict between the descriptive text contained in this Recommended Standard and the text of reference [9] or [10], the more detailed specifications contained in reference [9] or [10] are normative.

The COP fully specifies the closed-loop procedures executed by the sending and receiving ends of USLP. The COP consists of a pair of synchronized procedures for each VC: a FOP that executes within the sending entity, and a FARM that executes within the receiving entity. The sending FOP transmits Transfer Frames to the receiving FARM. The FARM returns to the FOP reports of the status of Transfer Frame acceptance using protocol control reports and thus closes the loop.

The inclusion of the COP provides a reliable QoS, i.e., the delivery of service data units to the layer above at the receiving end, correct and without omission or duplication, and in the same sequential order in which they were received from the layer above at the sending end.

For COP-1 over Telecommand Channel Coding in reference [6], correctness of the delivered service data units may be guaranteed (within known error probabilities) by the optional Frame Error Control Field (FECF) applied by the Data Link Protocol Sublayer, and by the Frame Validation Checks performed in USLP. For COP-P over Proximity-1 Channel Coding in reference [7], Frame Validation is carried out by the Proximity-1 Coding and Synchronization Sublayer, since the FECF is defined outside of the Version-3 Transfer Frame.

Validation of the completeness, sequentiality, and non-duplication of the delivered service data units on a particular Virtual Channel requires that an accounting (i.e., numbering) scheme for Transfer Frames be implemented by the COP.

Expedited Transfer Frames are processed by the COP only to the extent of causing the FARM to increment a counter for Expedited Transfer Frames. Expedited Transfer Frames are also used to send Control Commands from the FOP to the FARM.

2.1.2.5 Space Data Link Security Protocol (optional)

The Space Data Link Security (SDLS) protocol is specified in reference [14]. The SDLS protocol may provide security, such as authentication and encryption, for USLP Transfer Frames. Support for the SDLS protocol is an optional feature of USLP.

NOTE – The inclusion of the SDLS protocol makes no changes to any requirements in this Recommended Standard except for the inclusion of a mandatory Security Header and optional Security Trailer.

The inclusion of security provided by the SDLS protocol is selectable by VC. So, for example, there can be some VCs with security and some without. The type of security can vary from one VC to another.

2.1.3 ADDRESSING

There are four identifier fields in the header of Transfer Frames: Transfer Frame Version Number (TFVN), Spacecraft Identifier (SCID), Virtual Channel Identifier (VCID), and the Multiplexer Access Point Identifier (MAP ID). The concatenation of a TFVN and a SCID is

known as a Master Channel Identifier (MCID), and the concatenation of an MCID and a VCID is called a Global Virtual Channel Identifier (GVCID). The concatenation of the GVCID and a MAP ID is called the Global MAP ID (GMAP ID). Therefore

$$\text{MCID} = \text{TFVN} + \text{SCID};$$

$$\begin{aligned} \text{GVCID} &= \text{MCID} + \text{VCID} \\ &= \text{TFVN} + \text{SCID} + \text{VCID}. \end{aligned}$$

$$\begin{aligned} \text{GMAP ID} &= \text{GVCID} + \text{MAP ID} \\ &= \text{MCID} + \text{VCID} + \text{MAP ID} \\ &= \text{TFVN} + \text{SCID} + \text{VCID} + \text{MAP ID}. \end{aligned}$$

All Transfer Frames with the same MCID on a Physical Channel constitute a Master Channel. A Master Channel consists of one or more Virtual Channels. In most cases, a Physical Channel carries only Transfer Frames of a single MCID, and the Master Channel will be identical with the Physical Channel. However, a Physical Channel may carry Transfer Frames with multiple MCIDs (with the same TFVN, but different SCIDs). In such a case, the Physical Channel consists of multiple Master Channels. A Physical Channel is identified with a Physical Channel Name, which is set by management and not included in the header of Transfer Frames.

Each Virtual Channel in a Physical Channel is identified by a GVCID. Therefore a Virtual Channel consists of Transfer Frames with the same GVCID.

All Transfer Frames with the same GVCID and MAP ID constitute a MAP Channel. Thus a Virtual Channel consists of one or multiple MAP Channels.

The relationships between these Channels are shown in figure 2-2.

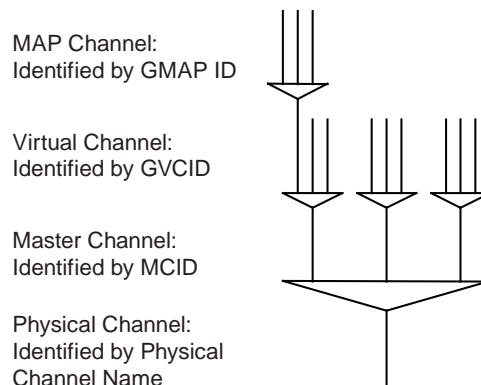


Figure 2-2: Relationships between Channels

2.1.4 PROTOCOL DESCRIPTION

USLP is described in terms of:

- a) the services provided to the users;
- b) the protocol data units; and
- c) the procedures performed by the protocol.

The service definitions are given in the form of primitives, which present an abstract model of the logical exchange of data and control information between the protocol entity and the service user. The definitions of primitives are independent of specific implementation approaches.

The procedure specifications define the procedures performed by protocol entities for the transfer of information between peer entities. The definitions of procedures are independent of specific implementation methods or technologies.

This protocol specification also specifies the requirements for the underlying services provided by the Synchronization and Channel Coding Sublayer.

2.2 OVERVIEW OF SERVICES

2.2.1 COMMON FEATURES OF SERVICES

USLP provides users with data transfer services. The point at which a service is provided by a protocol entity to a user is called a service access point (see reference [1]). Each service user is identified by a SAP address that is associated with a specific MAP ID within a specific Virtual Channel, i.e., the GMAP ID.

Service data units submitted to a SAP are processed in the order of submission. No processing order is maintained for service data units submitted to different SAPs.

NOTE – Implementations may be required to perform flow control at a 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.

The following are features common to all the services defined by this Recommended Standard:

- a) unidirectional (one way) services: One end of a connection can send, but not receive, data through the space link, while the other end can receive, but not send.
- b) asynchronous services: There are no predefined timing rules for the transfer of service data units supplied by the service user or for the transmission of Transfer Frames generated by the service provider. The user may request data transfer at any time, but there may be restrictions imposed by the service provider on the data generation rate. The timing of data transfer is determined by the provider in

accordance with mission-specific rules and may depend on the traffic of the data exchange by service provider/lower layers at the time of transfer.

- c) unconfirmed services: The sending user does not receive confirmation from the receiving end that data has been received.
- d) incomplete services: The services do not guarantee completeness, but some services may signal gaps in the sequence of service data units delivered to the receiving user.
- e) sequence-preserving services: The sequence of service data units supplied by the sending user is preserved through the transfer over the space link, although for the Expedited Service, described below, there may be gaps and duplications in the sequence of service data units delivered to the receiving user.

NOTE – This Recommended Standard assumes that these services are provided at the end points of a space link. However, this Recommended Standard makes no assumptions concerning how these end points are composed or configured either on board a spacecraft or in a ground system. In a ground system, the services defined by this Recommended Standard can be extended or enhanced with Space Link Extension Services (reference [E5]).

In addition, the following optional features are provided by these external CCSDS standards when selected by the user:

- a) The use of the COP procedures adds the capability for in-order delivery of service data units to the layer above without error or omission or duplication at the receiving end.
- b) The use of the SDLS protocol adds the capability to authenticate the sender and/or to encrypt the data to provide security for specific Virtual Channels defined to use the SDLS security services.

2.2.2 SERVICE ATTRIBUTES

2.2.2.1 Overview

USLP in association with the COP-1 (reference [9]), COP-P (reference [10]), and SDLS (reference [14]), provide the following service attributes (sequence-controlled, expedited, authenticated and/or encrypted, asynchronous, synchronous, and periodic) that determine the QoS associated with how reliably service data units supplied by the sending user are delivered to the receiving user.

The sequence-controlled QoS is provided by the COP-1 (reference [9]) or COP-P (reference [10]) procedures. The user requests with a parameter of the service request primitive whether the Sequence-Controlled or Expedited Service QoS should be applied to each service data unit.

The SDLS services provide the user with the capability of authenticating the Frame originator and/or encrypting the Frame contents. SDLS security services are established by the use of the managed parameters in section 6.

The capability to transfer data either in an asynchronous, synchronous, or periodic fashion is determined by the Transfer Frame multiplexing rules, which are specified by the missions.

These service attributes are provided at any service access point except for the Virtual Channel Frame and Master Channel Frame.

For the Virtual Channel Frame and Master Channel Frame Services, the service provider does not make any distinction between Sequence-Controlled and Expedited service QoS applicable to service data units supplied by the user. The user should request necessary procedures to provide Sequence-Controlled and/or Expedited Service QoS for its service data units.

2.2.2.2 Sequence-Controlled Service—Reliable Delivery

The reliable QoS (Sequence-Controlled Service) of either COP-1 (reference [9]) or COP-P (reference [10]) utilizes an Automatic Repeat Queuing (ARQ) procedure of the ‘go-back-*n*’ type with sequence-control mechanisms at both sending and receiving ends and a standard report returned from the receiving end to the sending end.

For Sequence-Controlled Service, service data units supplied by a sending user at a SAP are inserted into the Data Field of Transfer Frames (after MAP multiplexing when applicable) and transmitted on a Virtual Channel in the order in which they are presented at the SAP. The retransmission mechanism ensures with a high probability of success that:

- a) no service data unit is lost;
- b) no service data unit is duplicated;
- c) no service data unit is delivered out of sequence.

2.2.2.3 Expedited Service—Best Effort Delivery

The Best Effort QoS (Expedited Service) is used when ARQ is not required by the Data Link Layer, or when a higher layer protocol provides a retransmission capability.

For Expedited Service, service data units supplied by the sending user are transmitted one or more times (see reference [6]). There is no guarantee that all Expedited service data units are delivered to the receiving user.

NOTE – Although Expedited Service carries the name ‘Expedited’, it is neither a required method nor a faster method for sending urgent data to the receiving end.

2.2.2.4 Security Service

The optional use of the SDLS protocol (reference [14]) provides all its functions (authentication, encryption, authenticated encryption) for the data in the Transfer Frame Data Zone of a USLP Frame. It provides full protection for the service data of the MAP Packet (MAPP) service, the MAP Octet Stream Service, and the MAP Access (MAPA) Service (see section 6, Protocol Specification with SDLS option).

SDLS provides authentication for some fields in the Transfer Frame Primary Header and for some auxiliary data fields in a USLP Frame. It does not provide encryption for these fields.

SDLS provides no protection for the protocol control frames generated by either the COP-1 or COP-P.

SDLS provides no protection for the Master Channel Operational Control Field (MC_OCF) Service or for the Insert Service. It also provides no protection for the frames supplied to USLP by external sources such as by the Virtual Channel Frame (VCF) and the Master Channel Frame (MCF) services.

2.2.2.5 Asynchronous Service

In asynchronous service, there are no timing relationships between the transfer of service data units supplied by the service user and the transmission of Transfer Frames generated by the service provider. The user may request data transfer at any time it desires, but there may be restrictions imposed by the provider on the data generation rate. In this service (figure 2-3), each service data unit from a sending user is placed in a queue, the contents of which are sent to a receiving user in the order in which they were presented. Although transmission errors may prevent delivery of some data units, the service provider attempts to transfer all data units provided by the user exactly once. The timing of data transfer is determined by the provider in accordance with mission-specific rules and may depend on the traffic at the time of transfer. The key feature of this service is that all service data units from the sending user are transferred, and transferred only once.

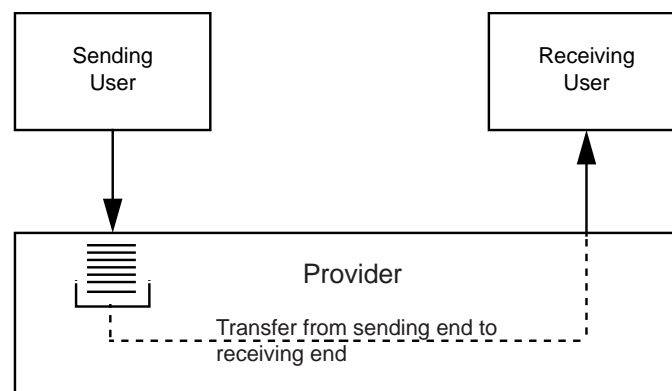


Figure 2-3: Asynchronous Service Model

2.2.2.6 Synchronous Service

In Synchronous Service (figure 2-4), each service data unit from a sending user is placed in a buffer that can hold only one service data unit; the content of the buffer is sent to a receiving user at the time when a Transfer Frame is transmitted. The transmission timing of Transfer Frames is determined by the transmission data rate and the fixed size of the Frame. The key feature of this service, which is essentially time-division multiplexing, is that the timing of data transfer is driven by the transfer mechanism, not by individual service requests from the user. Thus a particular service data unit from a user might be sent once, several times (if the 'new' value is not placed in the buffer soon enough), or not at all (if one value is replaced by a second before the service provider can send it).

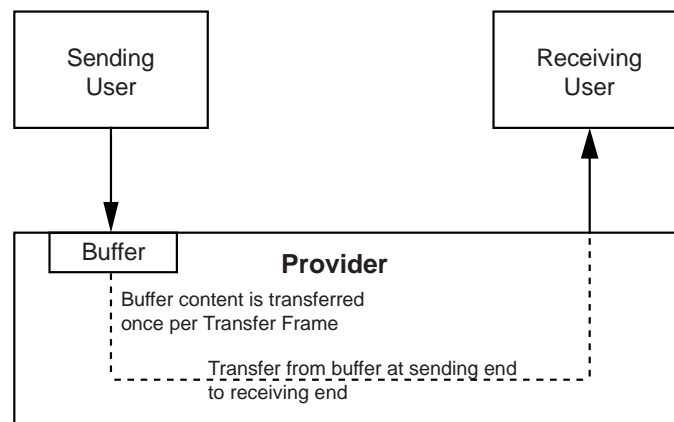


Figure 2-4: Synchronous Service Model

2.2.2.7 Periodic Service

Periodic service is a special case of synchronous service in which service data units are transferred at a constant rate. Periodic transfer from service interface to service interface is provided with a specified maximum delay and a specified maximum jitter at the service interface. There is one case in which periodic service is synchronous: when the service is associated with a Master Channel and that Master Channel produces Transfer Frames that are transmitted back-to-back.

For periodic services, all service data units are sent only once if the user supplies service data units at the same rate as the rate at which the service provider transfers them.

2.2.3 SUMMARY OF SERVICES

2.2.3.1 Introduction

Eight services are provided by USLP. Three of them (MAP Packet, MAP Access, and MAP Octet Stream) are provided for a MAP Channel. Two of them (Virtual Channel Frame and COPs Management) are provided for a Virtual Channel. Two of them (Master Channel Operational Control Field and Master Channel Frame) are provided for a Master Channel. One of them (Insert) is provided for all Transfer Frames on a Physical Channel.

Table 2-1 summarizes these services and shows their characteristics, the Service Data Units (SDUs) that they transfer, and the availability of SDLS security features. The optional SDLS protocol can provide security features for the SDUs transferred by some of the services:

- encryption, to provide confidentiality by hiding data content;
- authentication, to confirm the source and integrity of the data.

Table 2-1: Summary of Services Provided by Unified Space Data Link Protocol

Service	Service Attribute	Service Data Unit	SAP Address	SDLS Security Features
MAP Packet	Sequence-Controlled and Expedited, Asynchronous	Packet	GMAP ID + PVN	All
MAP Access	Sequence-Controlled and Expedited, Asynchronous or Periodic	MAPA_SDU	GMAP ID	All
MAP Octet Stream	Asynchronous or Periodic	Octet Stream Data	GMAP ID	All
Master Channel Operational Control Field	Synchronous or Asynchronous	OCF_SDU	GVCID	None
Virtual Channel Frame	Asynchronous or Synchronous	Transfer Frame	GVCID	None
Master Channel Frame	Asynchronous or Synchronous	Transfer Frame	MCID	None
Insert	Periodic	IN_SDU	Physical Channel Name	None
COPs Management	N/A	N/A	GVCID	N/A

2.2.3.2 MAP Packet Service

The MAP Packet Service transfers a sequence of variable-length, delimited, octet-aligned service data units known as Packets across a space link on a specified MAP Channel. The Packets transferred by this service must have a Packet Version Number (PVN) authorized by CCSDS. CCSDS PVNs are defined in reference [8].

The service is unidirectional and asynchronous. If the COP is used, then both Sequence-Controlled and Expedited QoS are provided for the MAP Packet Service.

When no reliable ARQ protocol is used, only Expedited service is provided for a MAP Channel. In this case, a user is identified with a single PVN and a GMAP ID.

2.2.3.3 MAP Access Service

The MAP Access Service provides transfer of a sequence of privately formatted service data units of variable length, called MAPA_SDUs, across a space link. The length of the service data units transferred by this service is not constrained by the length of the Data Field of the Transfer Frame. The MAP Access Service delivers MAPA_SDUs whose length is not included within the data unit itself.

For a given service instance, multiple users, each identified with the GMAP ID of the MAP Channel, can use this service on a MAP Channel. MAPA_SDUs from different users may be multiplexed together within one VC as long as these SDUs are multiplexed into the VC within the series of sequentially numbered Frames that contain each complete MAPA_SDU.

2.2.3.4 MAP Octet Stream Service

The MAP Octet Stream Service provides transfer of a serial string of aligned octets, whose internal structure and boundaries are unknown to the service provider, across a space link. The service is unidirectional, either asynchronous or periodic, and sequence-preserving. The service does not guarantee completeness, but it may signal gaps in the sequence of service data units delivered to the receiving user.

Octet Streams from different users may be multiplexed together within one Virtual Channel using different MAP IDs.

2.2.3.5 Master Channel Operational Control Field Service

The MC_OCF Service provides transfer of fixed-length data units, each consisting of four octets, in the Operational Control Field (OCF) of Transfer Frames of a Virtual Channel. The service is unidirectional and sequence-preserving. The transfer is signaled for inclusion within Transfer Frames of a Virtual Channel by using the OCF Flag in the Transfer Frame

Primary Header. The service does not guarantee completeness, but it may signal gaps in the sequence of service data units delivered to the receiving user.

For a given service instance, one or more users, identified with the GVCID of the Virtual Channel, can use this service on a Master Channel. Service data units from different users can be multiplexed together within one Master Channel.

2.2.3.6 Virtual Channel Frame Service

The Virtual Channel Frame Service provides transfer of a sequence of fixed- or variable-length USLP Transfer Frames of a Virtual Channel, created by an independent protocol entity, across a space link. The service is unidirectional, either synchronous or asynchronous, and sequence-preserving. The service does not guarantee completeness, but it may signal gaps in the sequence of service data units delivered to the receiving user.

For a given service instance, only one user, identified with the GVCID of the Virtual Channel, can use this service on a Virtual Channel. Service data units from different users are not multiplexed together within one Virtual Channel.

The Virtual Channel Frame Service transfers the independently created USLP Transfer Frames through a space link, together with USLP Transfer Frames created by the service provider itself. This service is made available to trusted users who are certified during the design process to ensure that the independently created protocol data units do not violate the operational integrity of the space link.

2.2.3.7 Master Channel Frame Service

The Master Channel Frame Service provides transfer of a sequence of fixed- or variable-length USLP Transfer Frames of a Master Channel, created by an independent protocol entity, across a space link. The service is unidirectional, either synchronous or asynchronous, and sequence-preserving. The service does not guarantee completeness, but it may signal gaps in the sequence of service data units delivered to a receiving user.

For a given service instance, only one user, identified with the MCID of the Master Channel, can use this service on a Master Channel. Service data units from different users are not multiplexed together within one Master Channel.

The Master Channel Frame Service transfers the independently created USLP Transfer Frames through the space link, together with USLP Transfer Frames created by the service provider itself. This service is made available to trusted users who are certified during the design process to ensure that the independently created protocol data units do not violate the operational integrity of the space link.

2.2.3.8 Insert Service

The Insert Service provides transfer of private, fixed-length, octet-aligned service data units in fixed-length Transfer Frames on the Physical Channel across a space link in a mode that efficiently utilizes the space link transmission resources. The service is unidirectional, periodic, and sequence-preserving. The service does not guarantee completeness, but may signal gaps in the sequence of service data units delivered to a receiving user.

For a given service instance, only one user, identified with the Physical Channel Name of the Physical Channel, can use this service on a Physical Channel. Service data units from different users are not multiplexed together within one Physical Channel. The presence of the Insert Zone is signaled by the Physical Channel Managed Parameters.

2.2.3.9 COPs Management Service

The COPs Management Service is used by a user at the sending end for managing the operations of either COP-1 or COP-P for a particular Virtual Channel. The user manages the operations of the COPs by invoking Directives defined in reference [9] for COP-1 and reference [10] for COP-P. The user is notified by the service provider of events associated with Directives and events that occur asynchronously with Directives.

A user of this service must be authorized to manage the COPs for a particular Virtual Channel. For a given service instance, only one user, identified with the GVCID of the Virtual Channel, is allowed to use this service on a Virtual Channel.

2.2.4 RESTRICTIONS ON SERVICES

There are some restrictions on the services provided on a Physical Channel, as follows:

- a) For fixed-length Transfer Frames only on one MAP Channel, the MAP Access Service cannot exist simultaneously with the MAP Packet Service.
- b) On one Master Channel, only one MC_OCF Service can exist.

2.3 OVERVIEW OF FUNCTIONS

2.3.1 GENERAL FUNCTIONS

USLP transfers various service data units, supplied by sending users, encapsulated in a sequence of protocol data units using services of lower layers. The protocol data units, known as USLP Transfer Frames, either have variable lengths and must be transferred over a Physical Channel asynchronously, or have a fixed length and must be transferred over a Physical Channel at a constant rate.

The protocol entity performs the following protocol functions:

- a) generation and processing of protocol control information (i.e., headers and trailers) to perform data identification, loss detection, and error detection;
- b) segmenting and blocking of service data units presented at the SAP in order to transfer variable-length service data units in either fixed- or variable-length protocol data units that are shorter than those service data units;
- c) multiplexing/demultiplexing and commutation/decommutation in order for various service users to share a single Physical Channel.

If the protocol entity supports the optional SDLS protocol, then it uses the functions of SDLS to apply the configured security features.

If the protocol entity supports the optional COP procedures, then it uses the functions of the COP to supply the sequence controlled features.

This protocol entity does not perform the following protocol functions:

- a) connection establishment and release;
- b) flow control;
- c) retransmission of protocol data units;
- d) management or configuration of the SDLS protocol.

2.3.2 INTERNAL ORGANIZATION OF PROTOCOL ENTITY

Figures 2-5 and 2-6 show the internal organization of the protocol entity of the sending and receiving ends, respectively. Data flow from top to bottom in figure 2-5 and from bottom to top in figure 2-6. These figures identify data-handling functions performed by the protocol entity and show logical relationships among these functions. The figures are not intended to imply any hardware or software configuration in a real system. Depending on the services actually used for a real system, not all of the functions may be present in the protocol entity.

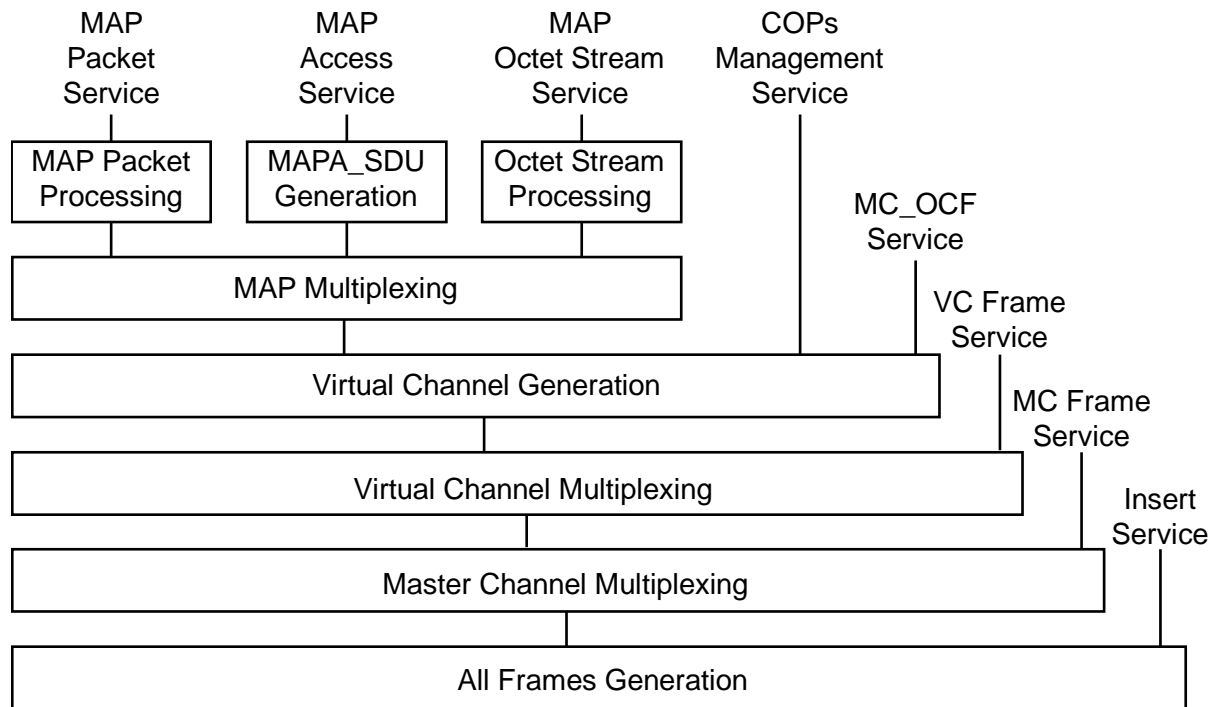


Figure 2-5: Internal Organization of Protocol Entity (Sending End)

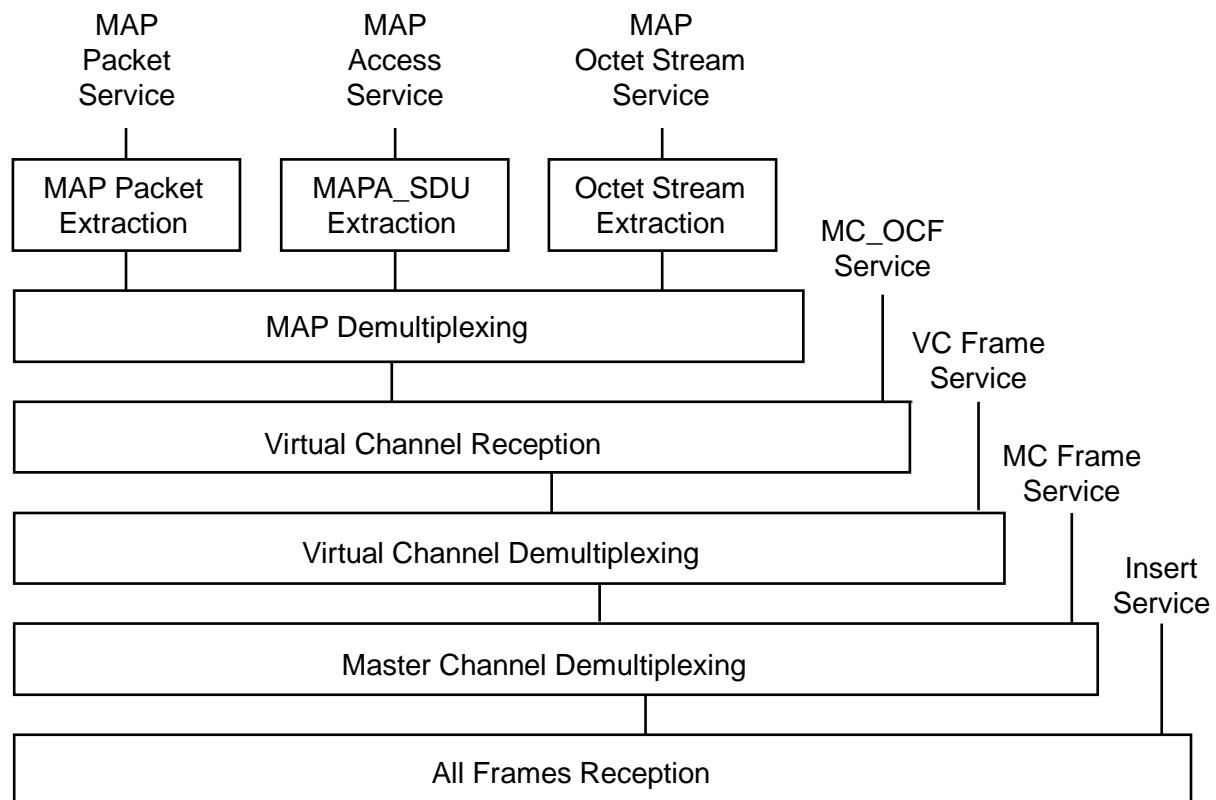


Figure 2-6: Internal Organization of Protocol Entity (Receiving End)

By extracting multiplexing/demultiplexing and commutation/decommutation functions from figures 2-5 and 2-6, the relationship among various data units can be shown as figure 2-7, which is known as the Channel Tree of USLP.

In figure 2-7, multiplexing (shown with a triangle) is a function of mixing, according to an algorithm established by the project, multiple streams of data units, each with a different identifier, to generate a single stream of data units. Commutation (shown with a box) is a function of concatenating, according to the formatting rule specified by the protocol definition, multiple data units, each from a different service, in a single protocol data unit sharing the same identifier.

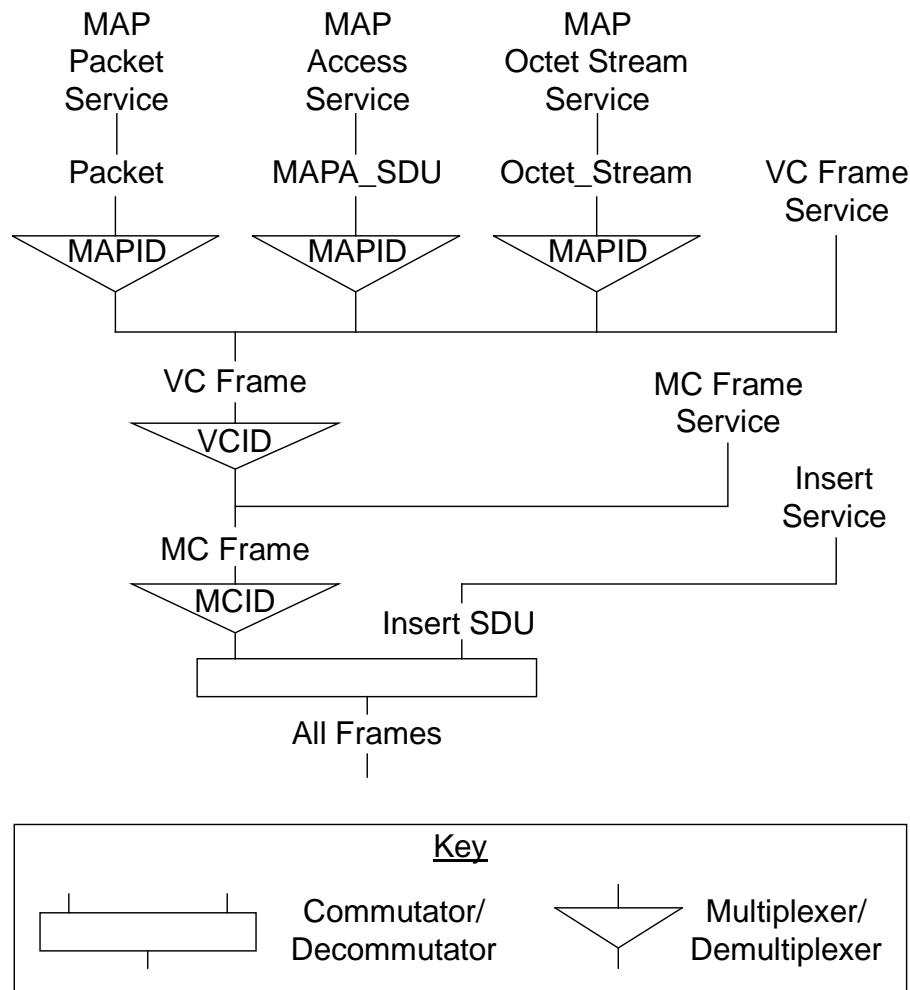


Figure 2-7: Unified Space Data Link Protocol Channel Tree

2.4 SERVICES ASSUMED FROM LOWER LAYERS

2.4.1 SERVICES ASSUMED FROM THE SYNCHRONIZATION AND CHANNEL CODING SUBLAYER

As described in 2.1.1, one of the set of Channel Coding and Synchronization Recommended Standards (references [3], [4], [5], [6], and [7]) are to be used with USLP with the constraints listed in this document as the Synchronization and Channel Coding Sublayer specification. The functions provided by the Synchronization and Channel Coding Recommended Standard are:

- a) error control encoding and decoding functions (optional when the coding schemes defined in references [3] through [7] are used);
- b) bit transition generation and removal functions (optional);
- c) delimiting and synchronizing functions;
- d) Frame validation function (see below for limitation).

When any of the coding schemes defined in references [3], [4], and [5] are used, the Synchronization and Channel Coding Sublayer transfers fixed-length, delimited Transfer Frames as a contiguous stream of bits over a space link using the services of the underlying Physical Layer.

When any of the coding schemes defined in references [6] and [7] are used, the Synchronization and Channel Coding Sublayer transfers variable-length, delimited Transfer Frames as an intermittent stream of bits over a space link using the services of the underlying Physical Layer.

Frame validation can be performed in the following ways depending upon the Channel Coding Sublayer specification selected:

- a) If any of the coding schemes defined in references [3], [4], and [5] are used, the TM Synchronization and Channel Coding Sublayer can deliver fully validated Frames with or without the use of the optional FECF.
- b) If any of the coding schemes defined in reference [6] are used, the TC Synchronization and Channel Coding Sublayer can deliver a fully validated data stream through use of the optional FECF, and USLP can deliver a fully validated Frame utilizing the Frame Delimiting and Fill Data Removal Procedure (4.3.9.2) and the USLP Frame Validation Check Procedure (4.3.9.3).
- c) If any of the coding schemes defined in reference [7] are used, the Proximity-1 Synchronization and Channel Coding Sublayer delivers fully validated Frames through use of the mandatory FECF.

For all coding schemes defined in references [3] through [7], one and only one Transfer Frame at a time is provided by the USLP Data Protocol Sublayer to the Synchronization and Channel Coding Sublayer.

2.4.2 SYSTEMATIC RETRANSMISSIONS

In addition, USLP, when reference [6] is applied, can request the Synchronization and Channel Coding Sublayer to perform systematic retransmissions of the data units submitted to it. The retransmissions can improve the probability of complete delivery for deep space missions on links with long light-time delays. This mechanism is not efficient but may be the best approach when frame reporting has significant lag time and thus reduces the efficiency of the COP-1 Go-Back-*N* retransmission process.

The definition of the service interface to the Synchronization and Channel Coding Sublayer specified in reference [6] includes the ChannelAccess.request service primitive, which has an optional Repetitions parameter. The sublayer transfers the data unit the number of times specified by Repetitions. If the value of Repetitions is one, or if the sublayer does not support the Repetitions parameter, then no systematic retransmissions are performed, and the Frame is transferred once.

USLP requests the systematic retransmissions in accordance with parameters set by management. For each MAP, management sets the value to be used for the Repetitions parameter when requesting the transfer of Frames carrying service data units on the Sequence-Controlled Service. For each Virtual Channel, management sets a similar parameter for Frames carrying COP control commands. For a Physical Channel, management sets an upper limit for the value of the Repetitions parameter specified in reference [6].

When requesting the transfer of Frames carrying service data units on the Expedited Service, USLP does not limit the value of the Repetitions parameter.

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 protocol entity 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 primitives. 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 can provide other parameters to the service user (e.g., parameters for controlling the service, monitoring performance, facilitating diagnosis, and so on).

3.2 SOURCE DATA

3.2.1 SOURCE DATA OVERVIEW

NOTE – This subsection describes the service data units that are transferred from sending users to receiving users by USLP.

The service data units transferred by USLP shall be:

- a) Packet;
- b) MAP Access Service Data Unit (MAPA_SDU);
- c) Octet Stream Data;
- d) Operational Control Field Service Data Unit (OCF_SDU);
- e) USLP Transfer Frame;
- f) Insert Service Data Unit (IN_SDU).

3.2.2 MAP PACKET SERVICE DATA UNIT

3.2.2.1 Packets shall be transferred over a space link with the MAP Packet Service.

3.2.2.2 The Packets transferred by this service must be self-delimiting and have a PVN registered by SANA (reference [8]).

3.2.2.3 The position and length of the Packet Length Field of the Packets must be known to the service provider in order to extract Packets from Transfer Frames at the receiving end.

NOTES

- 1 This requirement applies when packets span Transfer Frames using TFDZ Construction Rule '000' and also when blocking of Packets is performed by the service provider using TFDZ Construction Rule '111'. (See 4.1.4.2.2 TFDZ Construction Rules.)
- 2 Packets are variable-length, delimited, octet-aligned data units.
- 3 Examples of packets are: CCSDS Space Packets, CCSDS Encapsulation Packets.

3.2.2.4 Packets shall be contained either within a single Frame or within multiple sequential Frames of the same GVCID using the TFDZ Construction Rules in 4.1.4.2.2.

3.2.3 MAP ACCESS SERVICE DATA UNIT

3.2.3.1 MAPA_SDUs shall be transferred over a space link via the MAP Access Service.

3.2.3.2 A single MAPA_SDU may be transmitted in the Data Zone of one or multiple Transfer Frame(s), and therefore the length of MAPA_SDUs is not constrained by the length of the Transfer Frame Data Zone.

NOTE – MAPA_SDUs are variable-length, octet-aligned data units, the format of which is unknown to the service provider. Their length is provided to the SAP, and they are delimited within the TFDZ using the TFDZ construction rules.

3.2.4 MAP OCTET STREAM DATA

3.2.4.1 Octet Stream Data shall be transferred over a space link with the MAP Octet Stream Service.

3.2.4.2 The length of the Octet Stream Data supplied in each Octet Stream service request shall be used to delimit the received data that is to be transferred within the TFDZ.

NOTE – Octet Stream Data are variable-length, octet-aligned, continuous string of octets, the format of which is unknown to the service provider.

3.2.5 OPERATIONAL CONTROL FIELD SERVICE DATA UNIT

OCF_SDUs shall be transferred over a space link with the MC_OCF Service.

NOTES

- 1 Although the transfer of OCF_SDUs is carried within the Virtual Channel Frame, which by management can provide the OCF service, the creation of OCF_SDUs by

the sending user may or may not be synchronized with a single Virtual Channel. Such synchronization, if required for timing or other purposes, is a mission-design issue.

- 2 OCF_SDUs are fixed-length data units, each consisting of four octets, carried in the OCF, defined in 4.1.5, from a sending end to a receiving end.

3.2.6 USLP TRANSFER FRAME

The USLP Transfer Frame shall be used as the variable-length protocol data unit of USLP and may also be used as a service data unit of the Virtual Channel Frame and Master Channel Frame Services.

NOTE – The USLP Transfer Frame format is defined in 4.1 and 6.3 of this Recommended Standard.

3.2.7 INSERT SERVICE DATA UNIT

3.2.7.1 IN_SDUs shall be periodic, octet-aligned data units of fixed length. Their presence within the Frame shall be controlled by the Physical Channel Managed Parameter ‘Presence of Insert Zone’.

3.2.7.2 When present, each Transfer Frame on the Physical Channel shall contain one IN_SDU.

3.2.7.3 IN_SDU length shall be established by management and controlled by the Physical Channel Managed Parameter ‘Insert Zone Length’.

3.2.7.4 The length of the Insert Zone may be of any constant value that is an integral number of octets, between 1 octet and the maximum length of the data-carrying space of the Transfer Frame.

3.3 MAP PACKET SERVICE

3.3.1 OVERVIEW OF PACKET SERVICE

The MAP Packet Service transfers a sequence of variable-length, delimited, octet-aligned service data units known as Packets across a space link. The Packets transferred by this service must have a PVN recognized by CCSDS. Packet Version Numbers are registered in reference [8]. The service is unidirectional, asynchronous, and sequence-preserving. It does not guarantee completeness (because of potential gaps), nor does it signal gaps in the sequence of service data units delivered to a receiving user. A sequence discontinuity in Transfer Frames does not guarantee that there will be a gap in packets for a specific Application Processing ID (APID). (See reference [12].)

A user of this service is a protocol entity that sends or receives Packets with a single PVN and identified with the PVN and a GMAP ID. Different users (i.e., Packets with different PVNs) may share a single MAP Channel, and if there are multiple users on a MAP Channel, the service provider multiplexes Packets of different versions to form a single stream of Packets to be transferred on that MAP Channel.

3.3.2 MAP PACKET SERVICE PARAMETERS

3.3.2.1 General

The parameters used by the MAPP Service primitives shall conform to the specifications contained in subsections 3.3.2.2 through 3.3.2.9.

3.3.2.2 Packet

The Packet parameter shall contain a Packet for transfer by the MAP Packet Service.

NOTE – The Packet parameter is the service data unit transferred by the MAP Packet Service. Restrictions on the Packets transferred by the MAP Packet Service are stated in 3.2.2.

3.3.2.3 GMAP ID

The GMAP ID parameter shall contain a GMAP ID that indicates the MAP Channel through which the Packet is to be transferred.

NOTE – The GMAP ID is part of the SAP address of the MAP Packet Service.

3.3.2.4 Packet Version Number

The PVN shall identify the CCSDS protocol entity of the upper layer that uses the MAP Packet Service.

NOTE – The PVN is part of the SAP address of the MAP Packet Service.

3.3.2.5 SDU ID

The SDU ID parameter shall contain a user-supplied sequence number to be used to identify the associated Packet in subsequent MAPP_Notify.indication primitives.

3.3.2.6 QoS

At the sending end, the QoS parameter shall indicate whether the Packet should be transferred with the Sequence-Controlled QoS or the Expedited QoS.

3.3.2.7 Notification Type

In notifications to the user, the Notification Type parameter shall contain information about an event associated with the transfer of a Packet. The values taken by this parameter are defined in reference [9].

3.3.2.8 Packet Quality Indicator

The Packet Quality Indicator is an optional parameter that may be used to notify the user at the receiving end of the Packet Service whether the Packet delivered by the primitive is complete or partial.

3.3.2.9 Verification Status Code

3.3.2.9.1 The Verification Status Code is an optional parameter that may be used if the service provider supports the optional SDLS protocol.

3.3.2.9.2 The Verification Status Code parameter shall be used to notify the user at the receiving end of the Packet Service of a verification failure in a Transfer Frame addressed to the MAP Channel.

3.3.2.9.3 A non-zero value shall indicate that the SDLS protocol has detected an error; the values taken by this parameter are defined in reference [14].

NOTE – A non-zero value of the Verification Status Code does not indicate an error in the delivered Packet. Processing of Frames failing verification is implementation-specific and depends also on the processing capabilities of the service user for eventual forensic investigation.

3.3.3 MAP PACKET SERVICE PRIMITIVES

3.3.3.1 General

The service primitives associated with the MAPP service are:

- a) MAPP.request;
- b) MAPP_Notify.indication;
- c) MAPP.indication.

3.3.3.2 MAPP.request

3.3.3.2.1 Function

At the sending end, the MAPP Service user shall pass a MAPP.request primitive to the service provider to request that a Packet be transferred to the user at the receiving end through the specified MAP Channel.

3.3.3.2.2 Semantics

The MAPP.request primitive shall provide parameters as follows:

MAPP.request	(Packet, GMAP ID, Packet Version Number, SDU ID, QoS)
--------------	---

NOTE – When separate ports are provided for Sequence-Controlled and Expedited QoS, the QoS parameter is not used in this primitive.

3.3.3.2.3 When Generated

The sending-end user shall generate a MAPP.request primitive when a Packet is ready to be transferred.

3.3.3.2.4 Effect On Receipt

Receipt of the MAPP.request primitive shall cause the service provider to transfer the Packet.

3.3.3.3 MAPP_Notify.indication

3.3.3.3.1 Function

At the sending end, the service provider shall pass a MAPP_Notify.indication primitive to the MAPP Service user to notify the user of an event associated with the transfer of a Packet.

3.3.3.3.2 Semantics

The MAPP_Notify.indication primitive shall provide parameters as follows:

MAPP_Notify.indication	(GMAP ID, Packet Version Number, SDU ID, QoS, Notification Type)
------------------------	--

NOTE – When separate ports are provided for Sequence-Controlled and Expedited QoS, the QoS parameter is not used in this primitive.

3.3.3.3.3 When Generated

The sending-end service provider shall generate a MAPP_Notify.indication primitive in response to an event associated with the transfer of a Packet.

3.3.3.3.4 Effect On Receipt

The effect of receipt of the MAPP_Notify.indication primitive by the MAPP Service user is undefined.

3.3.3.4 MAPP.indication

3.3.3.4.1 Function

At the receiving end, the service provider shall pass a MAPP.indication to the MAPP Service user to deliver a Packet.

3.3.3.4.2 Semantics

The MAPP.indication primitive shall provide parameters as follows:

MAPP.indication	(Packet, GMAP ID, Packet Version Number, QoS (optional), Packet Quality Indicator (optional), Verification Status Code (optional))
-----------------	---

3.3.3.4.3 When Generated

The receiving-end service provider shall generate a MAPP.indication primitive when a Packet is ready to be delivered.

3.3.3.4.4 Effect On Receipt

The effect of receipt of the MAPP.indication primitive by the MAPP Service user is undefined.

3.4 MAP ACCESS SERVICE

3.4.1 OVERVIEW

The MAP Access Service provides transfer of a sequence of privately formatted, octet-aligned, variable-length service data units across a space link. The length of the data unit is unknown to the service provider and must be conveyed to the service provider at the service access point. The service is unidirectional, either asynchronous or periodic, and sequence-preserving. The service does not guarantee completeness, but is capable of signaling gaps in the sequence of service data units delivered to the receiving user, if the Virtual Channel is limited to one and only one MAP ID assignment.

A user of this service is a protocol entity that sends or receives a MAPA_SDU within one or more Frames identified with a USLP Protocol Identifier (UPID) and a GMAP ID. Different users (i.e., MAPA_SDUs identified with different UPIDs) may share a single MAP Channel but cannot share a single TFDZ. The service provider isolates each MAPA_SDU to a single Frame or sequentially numbered Frames using the TFDZ construction flags (see 4.1.4.2.2) on the same GMAP ID.

3.4.2 MAPA SERVICE PARAMETERS

3.4.2.1 General

The parameters used by the MAPA Service primitives shall conform to the specifications contained in subsections 3.4.2.2 through 3.4.2.7.

3.4.2.2 MAPA_SDU

The parameter MAPA_SDU shall be the service data unit transferred by the MAPA Service.

NOTE – Restrictions on the MAPA_SDUs transferred by the MAPA Service are stated in 3.2.3.

3.4.2.3 GMAP ID

The GMAP ID parameter shall contain a GMAP ID that indicates the MAP Channel through which the MAPA_SDU is to be transferred.

NOTE – The GMAP ID is the SAP address of the MAPA Service.

3.4.2.4 SDU ID

The SDU ID parameter shall contain a user-supplied sequence number to be used to identify the associated MAPA_SDU in subsequent MAPA_Notify.indication primitives.

3.4.2.5 QoS

At the sending end, the QoS parameter shall indicate whether the MAPA_SDU should be transferred with the Sequence-Controlled QoS or the Expedited QoS.

3.4.2.6 MAPA_SDU Loss Flag

3.4.2.6.1 The MAPA_SDU Loss Flag is an optional parameter that may be used to notify the user at the receiving end of the MAPA Service that a sequence discontinuity has been detected, and that one or more MAPA_SDUs have been lost.

3.4.2.6.2 Gaps may be reported only if the Virtual Channel is limited to one and only one MAP ID.

3.4.2.6.3 If implemented, the flag shall be derived by examining the Virtual Channel Frame Count in the Transfer Frames.

3.4.2.7 Verification Status Code

3.4.2.7.1 The Verification Status Code is an optional parameter that may be used if the service provider supports the optional SDLS protocol.

3.4.2.7.2 The parameter shall be used to notify the user at the receiving end of the MAPA Service of a verification failure in a Transfer Frame addressed to the Virtual Channel.

3.4.2.7.3 A non-zero value shall indicate that the SDLS protocol has detected an error: the values taken by this parameter are defined in reference [14].

NOTE – A non-zero value of the Verification Status Code does not indicate an error in the delivered MAPA_SDU. Processing of Frames failing verification is implementation-specific and depends also on the processing capabilities of the service user for eventual forensic investigation.

3.4.3 MAPA SERVICE PRIMITIVES

3.4.3.1 General

The service primitives associated with the MAPA service are:

- a) MAPA.request;
- b) MAPA_Notify.indication;
- c) MAPA.indication.

3.4.3.2 MAPA.request

3.4.3.2.1 Function

At the sending end, the MAPA Service user shall pass a MAPA.request primitive to the service provider to request that a MAPA_SDU be transferred to the user at the receiving end through the specified MAP Channel.

NOTE – The MAPA.request primitive is the service request primitive for the MAPA Service.

3.4.3.2.2 Semantics

The MAPA.request primitive shall provide parameters as follows:

MAPA.request	(MAPA_SDU, GMAP ID, SDU ID, QoS)
--------------	---

NOTE – When separate ports are provided for Sequence-Controlled and Expedited QoS, the QoS parameter is not used in this primitive.

3.4.3.2.3 When Generated

The sending-end service user shall generate a MAPA.request when a MAPA_SDU is ready to be transferred.

3.4.3.2.4 Effect On Receipt

Receipt of the MAPA.request primitive shall cause the service provider to transfer the MAPA_SDU.

3.4.3.3 MAPA_Notify.indication

3.4.3.3.1 Function

At the sending end, the service provider shall pass a MAPA_Notify.indication primitive to the MAPA Service user to notify the user of an event associated with the transfer of a MAPA_SDU.

3.4.3.3.2 Semantics

The MAPA.indication primitive shall provide parameters as follows:

MAPA_Notify.indication	(GMAP ID, SDU ID, QoS, Notification Type)
------------------------	---

NOTE – When separate ports are provided for Sequence-Controlled and Expedited QoS, the QoS parameter is not used in this primitive.

3.4.3.3.3 When Generated

The sending-end service provider shall generate a MAPA_Notify.indication in response to an event associated with the transfer of a MAPA_SDU.

3.4.3.3.4 Effect On Receipt

The effect of receipt of the MAPA_Notify.indication primitive by the MAPA Service user is undefined.

3.4.3.4 MAPA.indication

3.4.3.4.1 Function

At the receiving end, the service provider shall pass a MAPA.indication to the MAPA Service user to deliver a MAPA_SDU.

NOTE – The MAPA.indication primitive is the service indication primitive for the MAPA Service.

3.4.3.4.2 Semantics

The MAPA.indication primitive shall provide parameters as follows:

MAPA.indication	(MAPA_SDU, GMAP ID, QoS (optional), MAPA_SDU Loss Flag (optional), Verification Status Code (optional))
-----------------	--

3.4.3.4.3 When Generated

The receiving-end service provider shall generate a MAPA.indication when a MAPA_SDU is ready to be delivered.

3.4.3.4.4 Effect On Receipt

The effect of receipt of the MAPA.indication primitive by the MAPA Service user is undefined.

3.5 MAP OCTET STREAM SERVICE

3.5.1 OVERVIEW OF MAP OCTET STREAM SERVICE

The MAP Octet Stream Service provides transfer of a serial string of octets, whose internal structure and boundaries are unknown to the service provider, across a space link. The service is unidirectional, either asynchronous or periodic, and sequence-preserving. The service does not guarantee completeness, but may signal gaps in the sequence of service data units delivered to the receiving user.

Octet Streams from different users may be multiplexed together within one Virtual Channel using different MAP IDs.

3.5.2 MAP OCTET STREAM SERVICE PARAMETERS

3.5.2.1 General

The parameters used by the MAP Octet Stream Service primitives shall conform to the specifications contained in subsections 3.5.2.2 through 3.5.2.6.

3.5.2.2 Octet Stream Data

The parameter Octet Stream Data shall be the service data unit transferred by the MAP Octet Stream Service.

NOTE – Restrictions on the Octet Stream Data transferred by the MAP Octet Stream Service are stated in 3.2.3.

3.5.2.3 GMAP ID

The GMAP ID parameter shall contain a GMAP that indicates the MAP through which the Octet Stream Data is to be transferred.

NOTE – The GMAP ID is the SAP address of the MAP Octet Stream Service.

3.5.2.4 QoS

At the sending end, the QoS parameter shall indicate whether the Octet Stream Data should be transferred with the Sequence-Controlled QoS or the Expedited QoS.

3.5.2.5 Octet Stream Data Loss Flag

The Octet Stream Data Loss Flag is an optional parameter that may be used to notify the user at the receiving end of the MAP Octet Stream Service that a sequence discontinuity has been

detected and that some Octet Stream Data may have been lost. If implemented, the flag shall be derived by examining the Virtual Channel Frame Count in the Transfer Frames.

NOTE – As the contents of lost Transfer Frames cannot be established, the user should be aware that the Octet Stream Data Loss Flag signals a disruption in the Transfer Frames of the specified Virtual Channel, and not necessarily a disruption of the Octet Stream Data itself.

3.5.2.6 Verification Status Code

3.5.2.6.1 The Verification Status Code is an optional parameter that may be used if the service provider supports the optional SDLS protocol.

3.5.2.6.2 The Verification Status Code parameter shall be used to notify the user at the receiving end of the MAP Octet Stream Service of a verification failure in a Transfer Frame addressed to the Virtual Channel.

3.5.2.6.3 A non-zero value shall indicate that the SDLS protocol has detected an error: the values taken by this parameter are defined in reference [14].

NOTE – A non-zero value of the Verification Status Code does not indicate an error in the delivered Octet Stream Data. Processing of Frames failing verification is implementation-specific and depends also on the processing capabilities of the service user for eventual forensic investigation.

3.5.3 MAP OCTET STREAM SERVICE PRIMITIVES

3.5.3.1 General

The service primitives associated with the MAP Octet Stream Service are:

- a) OCTET_STREAM.request;
- b) OCTET_STREAM.indication.

3.5.3.2 OCTET_STREAM.request

3.5.3.2.1 Function

At the sending end, the MAP Octet Stream Service user shall pass an OCTET_STREAM.request primitive to the service provider to request that Octet Stream Data be transferred to the user at the receiving end through the specified Virtual Channel.

NOTE – The OCTET_STREAM.request primitive is the service request primitive for the MAP Octet Stream Service.

3.5.3.2.2 Semantics

The OCTET_STREAM.request primitive shall provide parameters as follows:

OCTET_STREAM.request (Octet Stream Data,
GMAP ID, QoS);

NOTE – When separate ports are provided for Sequence-Controlled and Expedited QoS, the QoS parameter is not used in this primitive.

3.5.3.2.3 When Generated

The sending-end service provider shall generate an OCTET_STREAM.request when Octet Stream Data is ready to be transferred.

3.5.3.2.4 Effect On Receipt

Receipt of the OCTET_STREAM.request primitive shall cause the service provider to transfer the Octet Stream Data.

NOTES

- 1 The OCTET_STREAM.request primitive is used to transfer Octet Stream Data across the space link on the specified Virtual Channel.
- 2 Since the service interface specification is an abstract specification, the implementation of the Octet Stream Data parameter is not constrained; i.e., it can be a continuous octet stream, a delimited octet stream, or individual bits.

3.5.3.3 OCTET_STREAM.indication

3.5.3.3.1 Function

At the receiving end, the service provider shall pass an OCTET_STREAM.indication to the MAP Octet Stream Service user to deliver Octet Stream Data.

NOTE – The OCTET_STREAM.indication primitive is the service indication primitive for the MAP Octet Stream Service.

3.5.3.3.2 Semantics

The OCTET_STREAM.indication primitive shall provide parameters as follows:

OCTET_STREAM.indication	(Octet Stream Data, GMAP ID, QoS (optional), Octet Stream Data Loss Flag (optional), Verification Status Code (optional))
-------------------------	--

3.5.3.3.3 When Generated

The receiving-end service provider shall generate an OCTET_STREAM.indication when Octet Stream Data is ready to be delivered.

3.5.3.3.4 Effect On Receipt

The effect of receipt of the OCTET_STREAM.indication primitive by the MAP Octet Stream Service user is undefined.

NOTES

- 1 The OCTET_STREAM.indication primitive is used to deliver Octet Stream Data to the MAP Octet Stream Service user identified by the GMAP ID.
- 2 The quantity of Octet Stream Data delivered by an implementation of this service primitive is not defined. Therefore it is not necessarily related to the quantity of Octet Stream Data submitted to the service provider by the sending user with the OCTET_STREAM.request primitive.

3.6 MASTER CHANNEL OPERATIONAL CONTROL FIELD SERVICE

3.6.1 OVERVIEW

The MC_OCF Service provides transfer of fixed-length data units, each consisting of four octets, in the OCF of Transfer Frames in a Virtual Channel. The service is unidirectional and sequence-preserving. The inclusion of an OCF_SDU in a Transfer Frame is signaled in the Transfer Frame Header. The transfer is synchronized with the release of Transfer Frames of a Virtual Channel. The service does not guarantee completeness, but it may signal gaps in the sequence of service data units delivered to the receiving user.

Only one user on the receive side, identified with the MCID of the Master Channel, can use the MC_OCF Service on a given Master Channel. Only one MC_OCF service can exist within a Master Channel, and that service may be utilized by multiple VCs that are designated by the VC Managed Parameter 'Inclusion of OCF' to support the service.

Only the COP and SDLS users can use this service on a Virtual Channel, because the OCF service can distinguish between their respective reports. The user is identified with the GVCID of the Virtual Channel. Service data units from different OCF users are multiplexed together for delivery within the OCF of the transmitted Virtual Channel.

3.6.2 MC_OCF SERVICE PARAMETERS

3.6.2.1 General

The parameters used by the MC_OCF Service primitives shall conform to the specifications contained in subsections 3.6.2.2 through 3.6.2.4.

3.6.2.2 OCF_SDU

The parameter OCF_SDU shall be the service data unit transferred by the MC_OCF Service in the Operational Control Field of Transfer Frames of a Virtual Channel.

NOTE – Restrictions on the OCF_SDU transferred by the MC_OCF Service are stated in 3.2.5.

3.6.2.3 GVCID

The GVCID parameter shall contain a GVCID that indicates the Virtual Channel through which the OCF_SDU is to be transferred.

NOTE – The GVCID is the SAP address of the MC_OCF Service.

3.6.2.4 OCF_SDU Loss Flag

3.6.2.4.1 The OCF_SDU Loss Flag is an optional parameter that may be used to notify the user at the receiving end of the MC_OCF Service that a sequence discontinuity has been detected and that one or more OCF_SDUs may have been lost.

3.6.2.4.2 If implemented, the flag shall be derived by examining the Virtual Channel Frame Count in the Transfer Frames.

3.6.3 MC_OCF SERVICE PRIMITIVES

3.6.3.1 General

The service primitives associated with this service are:

- a) MC_OCF.request;
- b) MC_OCF.indication.

3.6.3.2 MC_OCF.request

3.6.3.2.1 Function

At the sending end, the MC_OCF Service user shall pass a MC_OCF.request primitive to the service provider to request that an OCF_SDU be transferred to the user at the receiving end through the specified Virtual Channel.

NOTE – The MC_OCF.request primitive is the service request primitive for the MC_OCF Service.

3.6.3.2.2 Semantics

The MC_OCF.request primitive shall provide parameters as follows:

MC_OCF.request	(OCF_SDU, GVCID)
----------------	---------------------

3.6.3.2.3 When Generated

The MC_OCF.request primitive shall be passed to the service provider to request it to send the OCF_SDU.

3.6.3.2.4 Effect On Receipt

Receipt of the MC_OCF.request primitive shall cause the service provider to transfer the OCF_SDU.

NOTE – The MC_OCF.request primitive is used to transfer OCF_SDU across the space link on the specified Virtual Channel.

3.6.3.3 MC_OCF.indication

3.6.3.3.1 Function

At the receiving end, the service provider shall pass a MC_OCF.indication to the MC_OCF Service user to deliver an OCF_SDU.

NOTE – The MC_OCF.indication primitive is the service indication primitive for the MC_OCF Service.

3.6.3.3.2 Semantics

The MC_OCF.indication primitive shall provide parameters as follows:

MC_OCF.indication	(OCF_SDU, GVCID, OCF_SDU Loss Flag (optional))
-------------------	--

3.6.3.3.3 When Generated

The MC_OCF.indication primitive shall be passed from the service provider to the MC_OCF Service user at the receiving end to deliver an OCF_SDU.

3.6.3.3.4 Effect On Receipt

The effect of receipt of the MC_OCF.indication primitive by the MC_OCF Service user is undefined.

NOTE – The MC_OCF.indication primitive is used to deliver OCF_SDUs to the MC_OCF Service user identified by the GVCID.

3.7 VIRTUAL CHANNEL FRAME SERVICE

3.7.1 OVERVIEW OF VCF SERVICE

The VCF Service provides transfer of a sequence of variable-length USLP Transfer Frames of a Virtual Channel, created by an independent protocol entity, across a space link. The service is unidirectional, either asynchronous or periodic, and sequence-preserving. The service does not guarantee completeness, but it may signal gaps in the sequence of service data units delivered to the receiving user.

Only one user can use this service on a Virtual Channel, and the user is identified with the GVCID of the Virtual Channel. Service data units from different users are not multiplexed together within one Virtual Channel.

3.7.2 VCF SERVICE PARAMETERS

3.7.2.1 General

The parameters used by the VCF Service primitives shall conform to the specifications contained in subsections 3.7.2.2 through 3.7.2.4.

3.7.2.2 Frame

The Frame parameter shall be a USLP Transfer Frame of the Virtual Channel specified by the GVCID parameter.

NOTES

- 1 The Frame parameter is the service data unit transferred by the VCF Service.
- 2 The format of the GVCID parameter is defined in 4.1.

3.7.2.3 GVCID

The GVCID parameter shall contain a GVCID that indicates the Virtual Channel through which the Frame is to be transferred.

NOTE – The GVCID is the SAP address of the VCF Service.

3.7.2.4 Frame Loss Flag

3.7.2.4.1 The Frame Loss Flag is an optional parameter that may be used to notify the user at the receiving end of the VCF Service that a sequence discontinuity has been detected and that one or more Transfer Frames of the specified Virtual Channel have been lost.

3.7.2.4.2 If implemented, the flag shall be derived by examining the Virtual Channel Frame Count in the Transfer Frames.

3.7.3 VCF SERVICE PRIMITIVES

3.7.3.1 General

The service primitives associated with this service are:

- a) VCF.request;
- b) VCF.indication.

3.7.3.2 VCF.request

3.7.3.2.1 Function

At the sending end, the VCF Service user shall pass a VCF.request primitive to the service provider to request that a Frame be transferred to the user at the receiving end through the specified Virtual Channel.

NOTE – The VCF.request primitive is the service request primitive for the VCF Service.

3.7.3.2.2 Semantics

The VCF.request primitive shall provide parameters as follows:

VCF.request	(Frame, GVCID)
-------------	-------------------

3.7.3.2.3 When Generated

The VCF.request primitive shall be passed to the service provider to request it to send the Frame.

3.7.3.2.4 Effect On Receipt

Receipt of the VCF.request primitive shall cause the service provider to transfer the Frame.

NOTE – The VCF.request primitive is used to transfer Transfer Frames of a Virtual Channel across the space link.

3.7.3.3 VCF.indication

3.7.3.3.1 Function

At the receiving end, the service provider shall pass a VCF.indication to the VCF Service user to deliver a Frame.

NOTE – The VCF.indication primitive is the service indication primitive for the VCF Service.

3.7.3.3.2 Semantics

The VCF.indication primitive shall provide parameters as follows:

VCF.indication	(Frame, GVCID, Frame Loss Flag (optional))
----------------	--

3.7.3.3.3 When Generated

The VCF.indication primitive is passed from the service provider to the VCF Service user at the receiving end to deliver a Frame.

3.7.3.3.4 Effect On Receipt

The effect of receipt of the VCF.indication primitive by the VCF Service user is undefined.

NOTE – The VCF.indication primitive is used to deliver Transfer Frames of a Virtual Channel to the VCF Service user identified by the GVCID.

3.8 MASTER CHANNEL FRAME SERVICE

3.8.1 OVERVIEW OF MCF SERVICE

The MCF Service provides transfer of a sequence of fixed-length USLP Transfer Frames of a Master Channel, created by an independent protocol entity, across a space link. The service is unidirectional, either asynchronous or periodic, and sequence-preserving. The service does not guarantee completeness, but it may signal gaps in the sequence of service data units delivered to a receiving user.

Only one user can use this service on a Master Channel, and the user is identified with the MCID of the Master Channel. Service data units from different users are not multiplexed together within one Master Channel.

3.8.2 MCF SERVICE PARAMETERS

3.8.2.1 General

The parameters used by the MCF Service primitives shall conform to the specifications contained in subsections 3.8.2.2 through 3.8.2.4.

3.8.2.2 Frame

The Frame parameter shall be a USLP Transfer Frame of the Master Channel specified by the MCID parameter.

NOTES

- 1 The Frame parameter is the service data unit transferred by the VCF Service.
- 2 The format of the Frame parameter is defined in 4.1.
- 3 Formats of the USLP Transfer Frames transferred by the MCF Service are referenced in 3.2.6.

3.8.2.3 MCID

The MCID parameter shall contain the MCID of the Master Channel on which the Frame is to be transferred.

NOTE – The MCID is the SAP address of the MCF Service.

3.8.2.4 Frame Loss Flag

3.8.2.4.1 The Frame Loss Flag is an optional parameter that may be used to notify the user at the receiving end of the MCF Service that a sequence discontinuity has been detected and that one or more Transfer Frames of the specified Master Channel may have been lost.

3.8.2.4.2 If implemented, the flag shall be derived by a signal given by the underlying Synchronization and Channel Coding Sublayer.

3.8.3 MCF SERVICE PRIMITIVES

3.8.3.1 General

The service primitives associated with this service are:

- a) MCF.request;
- b) MCF.indication.

3.8.3.2 MCF.request

3.8.3.2.1 Function

At the sending end, the MCF Service user shall pass an MCF.request primitive to the service provider to request that a Frame be transferred to the user at the receiving end through the specified Master Channel.

NOTE – The MCF.request primitive is the service request primitive for the MCF Service.

3.8.3.2.2 Semantics

The MCF.request primitive shall provide parameters as follows:

MCF.request	(Frame, MCID)
-------------	------------------

3.8.3.2.3 When Generated

The MCF.request primitive shall be passed to the service provider to request it to send the Frame.

3.8.3.2.4 Effect On Receipt

Receipt of the MCF.request primitive shall cause the service provider to transfer the Frame.

NOTE – The MCF.request primitive is used to transfer Transfer Frames of a Master Channel across the space link.

3.8.3.3 MCF.indication

3.8.3.3.1 Function

At the receiving end, the service provider shall pass an MCF.indication to the MCF Service user to deliver a Frame.

NOTE – The MCF.indication primitive is the service indication primitive for the MCF Service.

3.8.3.3.2 Semantics

The MCF.indication primitive shall provide parameters as follows:

MCF.indication	(Frame, MCID, Frame Loss Flag (optional))
----------------	---

3.8.3.3.3 When Generated

The MCF.indication primitive shall be passed from the service provider to the MCF Service user at the receiving end to deliver a Frame.

3.8.3.3.4 Effect On Receipt

The effect of receipt of the MCF.indication primitive by the MCF Service user is undefined.

NOTE – The MCF.indication primitive is used to deliver Transfer Frames of a Master Channel to the VCF Service user identified by the MCID.

3.9 INSERT SERVICE

3.9.1 OVERVIEW OF INSERT SERVICE

The Insert Service provides transfer of private, fixed-length, octet-aligned service data units in fixed-length Transfer Frames across a space link in a mode that efficiently utilizes the space link transmission resources at relatively low data rates. The service is unidirectional, periodic, and sequence-preserving. The service does not guarantee completeness, but may signal gaps in the sequence of service data units delivered to a receiving user. The presence and length of the Insert Zone is controlled by the Physical Channel Managed Parameters.

Only one user can use this service on a Physical Channel, and the user is identified with the Physical Channel Name of the Physical Channel. Service data units from different users are not multiplexed together within one Physical Channel.

The Insert Service is provided only if the Physical Channel Managed Parameter ‘Transfer Frame Type’ is ‘Fixed Length’.

3.9.2 INSERT SERVICE PARAMETERS

3.9.2.1 General

The parameters used by the Insert Service primitives shall conform to the specifications contained in subsections 3.9.2.2 through 3.9.2.4.

3.9.2.2 IN_SDU

The IN_SDU parameter shall be the service data unit transferred by the Insert Service.

NOTE – Restrictions on the IN_SDUs transferred by the Insert Service are stated in 3.2.7.

3.9.2.3 Physical Channel Name

The Physical Channel Name shall indicate the Physical Channel through which the IN_SDU is to be transferred.

NOTE – The Physical Channel Name is the SAP address of the Insert Service.

3.9.2.4 IN_SDU Loss Flag

3.9.2.4.1 The IN_SDU Loss Flag is an optional parameter that may be used to notify the user at the receiving end of the Insert Service that a sequence discontinuity has been detected and that one or more IN_SDUs have been lost.

3.9.2.4.2 If implemented, the flag shall be derived by a signal given by the underlying Synchronization and Channel Coding Sublayer.

3.9.3 INSERT SERVICE PRIMITIVES

3.9.3.1 General

The service primitives associated with this service are:

- a) INSERT.request;
- b) INSERT.indication.

3.9.3.2 INSERT.request

3.9.3.2.1 Function

At the sending end, the Insert Service user shall pass an INSERT.request primitive to the service provider to request that an IN_SDU be transferred to the user at the receiving end through the specified Physical Channel.

NOTE – The INSERT.request primitive is the service request primitive for the Insert Service.

3.9.3.2.2 Semantics

The INSERT.request primitive shall provide parameters as follows:

INSERT.request	(IN_SDU, Physical Channel Name)
----------------	------------------------------------

3.9.3.2.3 When Generated

The INSERT.request primitive is passed to the service provider to request it to send the IN_SDU.

3.9.3.2.4 Effect On Receipt

Receipt of the INSERT.request primitive causes the service provider to transfer the IN_SDU.

NOTE – The INSERT.request primitive is used to transfer IN_SDU across the space link on the specified Physical Channel.

3.9.3.3 INSERT.indication

3.9.3.3.1 Function

At the receiving end, the service provider shall pass an INSERT.indication to the Insert Service user to deliver an IN_SDU.

NOTE – The INSERT.indication primitive is the service indication primitive for the Insert Service.

3.9.3.3.2 Semantics

The INSERT.indication primitive shall provide parameters as follows:

INSERT.indication	(IN_SDU, Physical Channel Name, IN_SDU Loss Flag (optional))
-------------------	--

3.9.3.3.3 When Generated

The INSERT.indication primitive shall be passed from the service provider to the Insert Service user at the receiving end to deliver an IN_SDU.

3.9.3.3.4 Effect On Receipt

The effect of receipt of the INSERT.indication primitive by the Insert Service user is undefined.

NOTE – The INSERT.indication primitive is used to deliver IN_SDUs to the Insert Service user identified by the Physical Channel Name.

-

3.10 COPS MANAGEMENT SERVICE

3.10.1 OVERVIEW

The COPS Management Service is used by a user at the sending end for managing the operations of either COP-1 or COP-P for a particular Virtual Channel or Port ID. The user manages the operations of COP by invoking Directives defined in reference [9] for COP-1 or in reference [10] for COP-P. The user is notified by the service provider of events associated with Directives and events that occur asynchronously with Directives.

A user of this service must be authorized to manage COP-1 for a particular Virtual Channel or COP-P for a particular Port ID. Only one user, identified either with the GVCID of the Virtual Channel for COP-1 or with an assigned Port ID for COP-P, is allowed to use this service.

3.10.2 COPS MANAGEMENT SERVICE PARAMETERS

3.10.2.1 General

The parameters used by the COPS Management Service primitives shall conform to the specifications of 3.10.2.2 through 3.10.2.7.

3.10.2.2 GVCID/Port ID

3.10.2.2.1 The GVCID parameter shall contain the GVCID of the Virtual Channel for which the COP-1 is managed.

NOTE – The GVCID consists of an MCID and a VCID and is the SAP address of the COP-1 Management Service.

3.10.2.2.2 The Port ID parameter shall contain the Port ID for which the COP-P is managed. (See reference [10].)

3.10.2.3 Directive ID

The Directive ID parameter shall contain a user-supplied sequence number to be used to identify the associated Directive.request primitive in subsequent Directive_Notify.indication primitives.

3.10.2.4 Directive Type

The Directive Type parameter shall contain the type of Directive.

NOTE – The values taken by this parameter are defined in reference [9] for COP-1 and reference [10] for COP-P.

3.10.2.5 Directive Qualifier

The Directive Qualifier parameter shall contain a qualifier of the Directive if one is required. The values taken by this parameter are defined in reference [9] for COP-1 and reference [10] for COP-P.

3.10.2.6 Notification Type

In notifications to the user, the Notification Type parameter shall contain information about an event associated with a Directive.

NOTE – The values taken by this parameter are defined in reference [9] for COP-1 and reference [10] for COP-P.

3.10.2.7 Notification Qualifier

The Notification Qualifier parameter shall contain a qualifier of the notification if one is required.

NOTE – The values taken by this parameter are defined in reference [9] for COP-1 and reference [10] for COP-P.

3.10.3 COPs MANAGEMENT SERVICE PRIMITIVES

3.10.3.1 General

The service primitives associated with the COPs Management Service are:

- a) Directive.request;
- b) Directive_Notify.indication;
- c) Async_Notify.indication.

3.10.3.2 Directive.request

3.10.3.2.1 Function

At the sending end, the authorized user shall pass a Directive.request primitive to the service provider to invoke a Directive defined in reference [9] for COP-1 or reference [10] for COP-P.

3.10.3.2.2 Semantics

The Directive.request primitive shall provide parameters as follows:

Directive.request	(GVCID/Port ID, Directive ID, Directive Type, Directive Qualifier)
-------------------	---

3.10.3.2.3 When Generated

The authorized user shall generate a Directive.request primitive when execution of a Directive is required.

3.10.3.2.4 Effect On Receipt

Receipt of the Directive.request primitive shall cause the service provider to execute the Directive.

3.10.3.3 Directive_Notify.indication

3.10.3.3.1 Function

At the sending end, the service provider shall pass a Directive_Notify.indication primitive to the authorized user to notify the user of an event or an action associated with a Directive requested by the user.

3.10.3.3.2 Semantics

The Directive_Notify.indication primitive shall provide parameters as follows:

Directive_Notify.indication (GVCID/Port ID,
Directive ID,
Notification Type)

3.10.3.3.3 When Generated

The sending-end service provider shall generate a Directive_Notify.indication primitive in response to an event or action associated with a Directive.

3.10.3.3.4 Effect On Receipt

The effect of receipt of the Directive_Notify.indication primitive by the COPs Management Service user is undefined.

3.10.3.4 Async_Notify.indication

3.10.3.4.1 Function

At the sending end, the service provider shall pass an Async_Notify.indication primitive to the authorized user to notify the user of an event that occurs asynchronously with requests.

3.10.3.4.2 Semantics

The Async_Notify.indication primitive shall provide parameters as follows:

Async_Notify.indication	(GVCID/Port ID, Notification Type, Notification Qualifier)
-------------------------	--

3.10.3.4.3 When Generated

The sending-end service provider shall generate an Async_Notify.indication primitive in response to an event that occurs asynchronously with requests.

3.10.3.4.4 Effect On Receipt

The effect of receipt of the Async_Notify.indication primitive by the COPs Management Service user is undefined.

4 PROTOCOL SPECIFICATION WITHOUT SDLS OPTION

NOTE – This section specifies the protocol data unit and the procedures of USLP without support for the SDLS protocol. Section 6 specifies the protocol with the SDLS option.

4.1 PROTOCOL DATA UNIT

4.1.1 USLP TRANSFER FRAME

A USLP Transfer Frame shall encompass the major fields, positioned contiguously, in the following sequence:

- a) Transfer Frame Primary Header (4 to 14 octets, mandatory);
- b) Transfer Frame Insert Zone (integral number of octets, optional);
- c) Transfer Frame Data Field (integral number of octets, optional);
- d) Operational Control Field (4 octets, optional);
- e) Frame Error Control Field (2 or 4 octets, optional).

NOTE – The structural components of the USLP Transfer Frame are shown in figure 4-1.

←----- USLP Transfer Frame ----->

Transfer Frame Primary Header	Transfer Frame Insert Zone (Optional)	Transfer Frame Data Field (Optional)		Operational Control Field (Optional)	Frame Error Control Field (Optional)
		Transfer Frame Data Field Header	Transfer Frame Data Zone (TFDZ)		
4–14 Octets	Varies	1–3 Octets	Varies	4 Octets	2 or 4 Octets

Figure 4-1: USLP Transfer Frame Structural Components

4.1.2 TRANSFER FRAME PRIMARY HEADER

4.1.2.1 General

The Transfer Frame Primary Header is mandatory and shall consist of up to 13 fields, positioned contiguously, in the following sequence:

- a) Transfer Frame Version Number (4 bits, mandatory);

- b) Spacecraft Identifier (16 bits, mandatory);
- c) Source or Destination Identifier (1 bit, mandatory);
- d) Virtual Channel Identifier (6 bits, mandatory);
- e) Multiplexer Access Point Identifier (4 bits, mandatory);
- f) End of Frame Primary Header Flag (1 bit, mandatory);
- g) Frame Length (16 bits, mandatory);
- h) Bypass/Sequence Control Flag (1 bit, mandatory);
- i) Protocol Control Command Flag (1 bit, mandatory);
- j) Reserve Spares (2 bits, mandatory);
- k) Operational Control Field Flag (1 bit, mandatory);
- l) Virtual Channel Frame Count Length (3 bits, mandatory);
- m) Virtual Channel Frame Count (0 to 7 octets, optional).

NOTE – The format of the non-truncated Transfer Frame Primary Header is shown in figure 4-2.

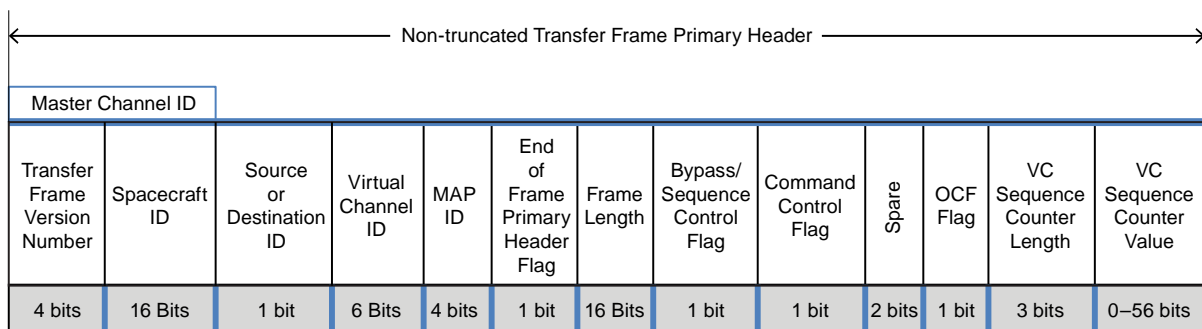


Figure 4-2: Non-truncated Transfer Frame Primary Header

4.1.2.2 Master Channel Identifier

4.1.2.2.1 General

4.1.2.2.1.1 Bits 0–19 of the Transfer Frame Primary Header shall contain the MCID.

4.1.2.2.1.2 The Master Channel Identifier shall consist of:

- a) Transfer Frame Version Number (4 bits, mandatory);
- b) Spacecraft Identifier (16 bits, mandatory).

4.1.2.2.2 Transfer Frame Version Number

4.1.2.2.2.1 Bits 0–3 of the Transfer Frame Primary Header shall contain the (Binary Encoded) Transfer Frame Version Number.

4.1.2.2.2.2 This 4-bit field shall identify the data unit as a Transfer Frame defined by this Recommended Standard; it shall be set to ‘1100’ binary.

NOTE – This Recommended Standard defines the USLP Version-4 Transfer Frame.

4.1.2.2.3 Spacecraft Identifier

Bits 4–19 of the Transfer Frame Primary Header shall contain the Spacecraft Identifier.

NOTE – SANA assigns Spacecraft Identifiers according to the procedures in reference [11].

4.1.2.3 Source-or-Destination Identifier

4.1.2.3.1 Bit 20 of the Transfer Frame Primary Header shall contain the Source-or-Destination Identifier.

4.1.2.3.2 The Source-or-Destination Identifier shall be used to identify the association of the data contained in the Transfer Frame.

4.1.2.3.3 The Source-or-Destination Identifier is interpreted as follows:

- a) ‘0’ = SCID refers to the *source* of the Transfer Frame;
- b) ‘1’ = SCID refers to the *destination* of the Transfer Frame.

NOTE – Although not limited to proximity environments, a discussion of the assignment of source vs. destination Spacecraft ID is provided in reference [E14].

4.1.2.4 Virtual Channel Identifier

4.1.2.4.1 Bits 21–26 of the Transfer Frame Primary Header shall contain the VCID.

4.1.2.4.2 The VCID shall be used to identify the Virtual Channel.

NOTE – There are no restrictions on the selection of VCIDs 0–62; in particular, Virtual Channels are not required to be numbered consecutively. VCID 63 is reserved for OID Transfer Frames (see 4.1.4.1.6 and 4.1.4.1.7).

4.1.2.5 Multiplexer Access Point Identifier

4.1.2.5.1 Bits 27–30 of the Transfer Frame Primary Header shall contain the MAP ID.

4.1.2.5.2 If only one MAP exists, then the MAP ID shall be set to a constant value for all data placed into the Transfer Frame Data Zone for that VC.

4.1.2.5.3 When multiple MAPs are in use on a Virtual Channel,

- a) a given SDU shall be assigned to only one MAP at a time;
- b) the entire SDU must be completed before a new SDU associated with any other MAP is transferred on that VC.

NOTES

- 1 The MAP Identifier provides the identification from 1 to up to 16 possible MAP Channels within a Virtual Channel.
- 2 There are no restrictions on the selection of MAPs. In particular, MAPs are not required to be numbered consecutively.
- 3 MAPs identify service access points for the Virtual Channel and may be designated to receive Sequence-Controlled or Expedited SDUs.

4.1.2.6 End of Frame Primary Header Flag

4.1.2.6.1 Bit 31 of the Transfer Frame Primary Header shall contain the End of Frame Primary Header Flag.

4.1.2.6.2 The End of Frame Primary Header Flag shall signal the use of the truncated Transfer Frame Primary Header (figure 4-3) in lieu of the complete header. The End of Frame Primary Header Flag is interpreted as follows:

- a) '0' = Transfer Frame Primary Header is not truncated and consists of up to the 13 fields defined in 4.1.2.1.
- b) '1' = The Transfer Frame Primary Header is truncated and is defined as containing only the first contiguous six fields of the Transfer Frame Primary Header. (See figure 4-3, Truncated Transfer Frame Primary Header.)

NOTE – The format of the truncated Transfer Frame Primary Header is shown in figure 4-3.

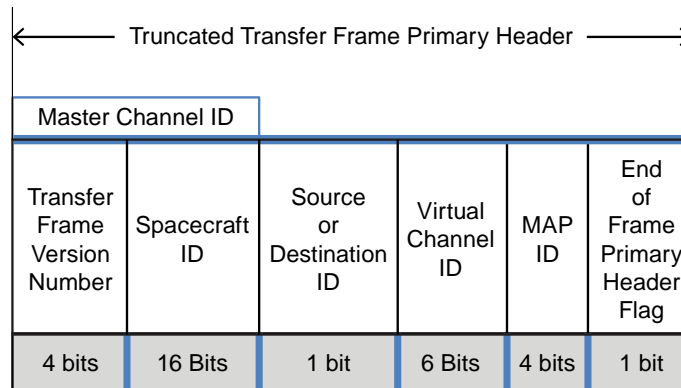


Figure 4-3: Truncated Transfer Frame Primary Header

4.1.2.6.3 The Transfer Frame shall contain either a non-truncated (see figure 4-2) or a truncated (see figure 4-3) Transfer Frame Primary Header.

4.1.2.6.4 When the truncated Transfer Frame Primary Header is used, the Transfer Frame is of fixed length constrained by the VC Managed Parameter ‘Truncated Transfer Frame Length’.

NOTE – The truncated Transfer Frame does not contain an Insert Zone, OCF, or FECF.

4.1.2.7 Frame Length

4.1.2.7.1 Bits 32–47 of the Transfer Frame Primary Header shall contain the Frame Length.

4.1.2.7.2 This 16-bit field shall contain a length count C that equals one fewer than the total octets in the Transfer Frame.

4.1.2.7.3 The count shall be measured from the first bit of the Transfer Frame Primary Header to the last bit of the Frame Error Control Field (if present), or to the last bit of the Operational Control Field (if present), or to the last bit of the Transfer Frame Data Field (if both the FECF and the OCF are omitted).

NOTES

- 1 If the Physical Channel Managed Parameter ‘Transfer Frame Type’ is ‘Fixed Length’ and the Transfer Frame is aligned to the codeblock, then the Maximum Transfer Frame Length is constrained by the specifications contained in references [3], [4], and [5].
- 2 The Maximum Transfer Frame Length of a variable-length Frame or an uncoded fixed-length Frame is constrained only by the size of the Frame Length field.
- 3 The length count C is expressed as:

$$C = (\text{Total Number of Octets in the Transfer Frame}) - 1$$

- 4 The size of this field limits the total number of octets in the Transfer Frame to 65536 octets.
- 5 In general, Transfer Frame Type can be either 'Fixed Length' or 'Variable Length' for a given Physical, Master, or Virtual Channel. If the Transfer Frame Type is 'Fixed Length' for the Physical Channel, then it is fixed for both the subordinate Master and Virtual Channels.

4.1.2.8 Bypass/Sequence Control Flag and Protocol Control Command Flag

4.1.2.8.1 Bypass/Sequence Control Flag

4.1.2.8.1.1 Bit 48 of the Transfer Frame Primary Header shall contain the Bypass/Sequence Control Flag.

4.1.2.8.1.2 The single-bit Bypass/Sequence Control Flag shall be used to control the application of Frame Acceptance Checks within the COP-1 (reference [9]) and COP-P (reference [10]) procedures by the receiving end:

- a) Sequence-Controlled QoS: setting the Bypass/Sequence Control Flag to value '0' shall specify that this Transfer Frame is a Sequence-Controlled Transfer Frame, and acceptance of this Transfer Frame by the receiving end shall be subject to the normal Frame Acceptance Checks of the FARM;
- b) Expedited QoS: setting the Bypass/Sequence Control Flag to value '1' shall specify that this Transfer Frame is an Expedited Transfer Frame, and the Frame Acceptance Checks of the FARM by the receiving end shall be bypassed.

NOTES

- 1 This field is equivalent to the QoS Indicator in reference [10].
- 2 The FARM associated with the COP-1 or COP-P can be made to operate in a normal Acceptance/Sequence-Controlled mode (for Sequence-Controlled Transfer Frames) or an Expedited mode (for Expedited Transfer Frames), according to the setting of the Bypass/Sequence Control Flag.
- 3 For COP-1 over Telecommand Channel Coding in reference [6], all Transfer Frames received by the receiving end first undergo a basic set of USLP Frame and TC Channel Coding Validation Checks, which are applied regardless of the setting of the Bypass/Sequence Control Flag (see 4.3.9.3).
- 4 For COP-P over Proximity-1 Channel Coding in reference [7], Frame Validation is carried out by the Proximity-1 Coding and Synchronization sublayer, since the FECF is defined outside of the Version-3 Transfer Frame.

4.1.2.8.2 Protocol Control Command Flag

4.1.2.8.2.1 Bit 49 of the Transfer Frame Primary Header shall contain the Protocol Control Command Flag.

4.1.2.8.2.2 The single-bit Protocol Control Command Flag shall be used to specify whether the Transfer Frame Data Field is conveying Protocol Control Commands or user data:

- a) setting the Protocol Control Command Flag to value '0' shall indicate that the Transfer Frame Data Field contains user data;
- b) setting the Protocol Control Command Flag to value '1' shall indicate that the Transfer Frame Data Field contains protocol control information.

NOTE – This field is equivalent to the PDU Type in reference [10].

4.1.2.8.3 Combined States of the Bypass Flag and Protocol Control Command Flag

The combined states of the Bypass Flag and Protocol Control Command Flag shall be interpreted by the receiving end as shown in table 4-1.

Table 4-1: Interpretation of the Bypass/Sequence Control and Protocol Control Command Flags

Bypass/ Sequence- Control Flag	Protocol Control Command Flag	Interpretation
0	0	Transfer Frame Data Field carries user data, subject to Frame Acceptance Checks under control of the FARM. These Frames use the AD (Sequence-Controlled) Service of the COP(-1/-P).
0	1	Reserved for future application.
1	0	Transfer Frame Data Field carries user data, with Frame Acceptance Checks bypassed under control of the FARM. These Frames use the BD (Expedited) Service of the COP(-1/-P).
1	1	Transfer Frame Data Field carries Protocol Control Commands, with Frame Acceptance Checks bypassed under control of the FARM. These Frames control the BC (Sequence-Controlled) Service of the COP(-1/-P).

4.1.2.9 Reserve Spares

4.1.2.9.1 Bits 50–51 of the Transfer Frame Primary Header shall contain the reserved spares.

4.1.2.9.2 This 2-bit field is reserved for future definition by CCSDS and shall be set to ‘00’.

4.1.2.10 Operational Control Field Flag

4.1.2.10.1 Bit 52 of the Transfer Frame Primary Header shall contain the Operational Control Field Flag.

4.1.2.10.2 The Operational Control Field Flag shall indicate the presence or absence of the Operational Control Field. It shall be ‘1’ if the Operational Control Field is present; it shall be ‘0’ if the Operational Control Field is not present.

4.1.2.11 Virtual Channel Frame Count Length

4.1.2.11.1 Bits 53–55 of the Transfer Frame Primary Header shall contain the length of the Virtual Channel Frame Count field.

4.1.2.11.2 This 3-bit field shall define the length of the Virtual Channel Frame Count field.

4.1.2.11.3 The value of this field shall be interpreted as shown in table 4-2.

Table 4-2: Interpretation of the Virtual Channel Frame Count Length

Value of VC Frame Count Length (binary)	Length of Virtual Channel Frame Count Field	Maximum Virtual Channel Frame Count
000	0	N/A
001	1 octet	255
010	2 octets	65,535
011	3 octets	16,777,215
100	4 octets	4,294,967,295
101	5 octets	1,099,511,627,775
110	6 octets	281,474,976,710,655
111	7 octets	7.20575940379279E16

4.1.2.11.4 The Virtual Channel Frame Count length for a given Virtual Channel is invariant.

NOTES

- 1 The VC Frame Count Length of ‘000’ indicates that the VC Frame Count field is absent for the Transfer Frame Header. This setting can be used for Frames using Expedited QoS.
- 2 The Frame Count Length is VC-dependent; i.e., this protocol maintains a separate Frame Count for each of the Virtual Channels.

- 3 The maximum Virtual Channel Frame Count is constrained by the size of the largest Expedited or Sequence-Controlled Transfer Frame count for that VC.

4.1.2.12 Virtual Channel Frame Count

4.1.2.12.1 If present, the Virtual Channel Frame Count shall be the final field in the Transfer Frame Header starting in bit 56.

NOTE – This field is optional and its size is signaled in the Virtual Channel Frame Count Length field.

4.1.2.12.2 The VC Frame Count shall contain a sequential binary count, i.e., modulo (maximum Virtual Channel Frame Count +1) of each Sequence-Controlled Transfer Frame transmitted within a specific Virtual Channel (see table 4-2).

4.1.2.12.3 The VC Frame Count shall be independent per VC and shall increment by 1 for each Sequence-Controlled Frame sent; i.e., the Bypass/Sequence Control Flag is set to '0'. In this case, the Frame Count is called the Sequence-Controlled Counter.

4.1.2.12.4 The VC Frame Count shall be independent per VC and shall increment by 1 for each Expedited Frame sent; i.e., the Bypass/Sequence Control Flag is set to '1'. In this case, the Frame Count is called the FARM-B (Expedited Frame) Counter.

4.1.2.12.5 The Virtual Channel Frame Count should not be reset before reaching the maximum Virtual Channel Frame Count.

NOTES

- 1 The purpose of this field is to provide individual accountability for each Virtual Channel. It verifies the sequentiality of Frames within a VC, thus enabling systematic concatenation of SDUs that extend across Frame boundaries. It also enables verification of the sequentiality of incoming Sequence-Controlled Frames by the FARM.
- 2 The procedure for assigning the VC Frame Count (i.e., Frame Sequence Number) to Transfer Frames is defined for COP-1 in reference [9] and for COP-P in reference [10].
- 3 If the Virtual Channel Frame Count is reset because of an unavoidable re-initialization, neither the completeness of a sequence of Transfer Frames in the related Virtual Channel can be determined nor the reassembly of SDUs can be guaranteed.
- 4 TC and Proximity-1 both require a sequence control count as well as an expedited count for the same VC.

- 5 USLP allows the user to define up to 63 VCs (VCID 63 is reserved). Constraining a VC to exclusively support either Sequence-Controlled or Expedited Service means that each VC would need only 1 VC counter.

4.1.3 TRANSFER FRAME INSERT ZONE

4.1.3.1 If present, the Transfer Frame Insert Zone shall follow, without gap, the Transfer Frame Primary Header.

4.1.3.2 If the Physical Channel Managed Parameter ‘Presence of Insert Zone’ is set to ‘Present’, then the Insert Zone shall exist in every Transfer Frame transmitted within the same Physical Channel, whose Physical Channel Managed Parameter ‘Frame Type’ equals ‘Fixed Length’, including Only Idle Data (OID) Transfer Frames.

4.1.3.3 For Insert Service, the length of the Insert Zone shall be set by management to be equal to the constant length of the IN_SDU for that Physical Channel. The Insert Zone shall contain precisely one octet-aligned IN_SDU.

4.1.3.4 Once set by management, the length of the Insert Zone shall be static throughout a Mission Phase.

NOTE – The presence, absence, and length of the Insert Zone is controlled by the Physical Channel Managed Parameters.

4.1.4 TRANSFER FRAME DATA FIELD

4.1.4.1 General

4.1.4.1.1 If present, the TFDF shall follow, without gap, the Transfer Frame Primary Header or the Transfer Frame Insert Zone, if present.

NOTE – If the MC_OCF service is the only service used on the channel or the truncated Transfer Frame Primary Header is in use, then the Transfer Frame does not contain a TFDF. (See 3.6, Master Channel_OCF Service and 4.1.2.6, End of Frame Primary Header Flag.)

4.1.4.1.2 The Transfer Frame Data Field shall

- a) contain an integer number of octets;
- b) be of variable length equal to the Frame Length minus the length of the Transfer Frame Primary Header minus the length of the Transfer Frame Insert Zone and/or the Operational Control Field and/or Frame Error Control Field (if any of these are present).

4.1.4.1.3 If present, the Transfer Frame Data Field shall consist of a TFDF Header followed by a Transfer Frame Data Zone (TFDZ) (see figure 4-4 below).

4.1.4.1.4 The Transfer Frame Data Field, which shall contain an integral number of octets, may vary in length up to a maximum of 65514 octets.

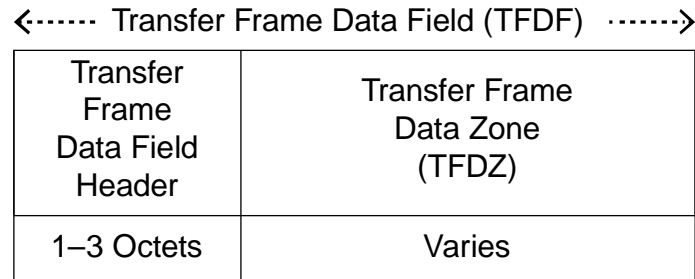


Figure 4-4: Transfer Frame Data Field

NOTE – The length of the TFDZ associated with a VC may be restricted by the associated VC Managed Parameters to be either fixed or variable (see 5.4).

4.1.4.1.5 In the case where no valid TFDF is available for transmission at release time for a given Virtual Channel, and the Physical Channel Managed Parameter ‘Transfer Frame Type’ equals ‘Fixed Length’ and the Physical Channel Managed Parameter ‘Generate OID Frame’ equals ‘True’, a Transfer Frame with a TFDZ containing Only Idle Data shall be transmitted.

NOTE – Such a Transfer Frame is called an OID Transfer Frame.

4.1.4.1.6 The Virtual Channel ID of an OID Transfer Frame shall be set to the value of ‘all ones’ (i.e., 63 decimal).

4.1.4.1.7 VCID 63 shall be the only Virtual Channel used for OID Transfer Frame transmission.

4.1.4.1.8 The MAP ID of an OID Transfer Frame shall be set to the value of ‘0’.

4.1.4.1.9 The Transfer Frame Data Field Header of an OID Transfer Frame shall consist of the TFDZ Construction Rule set to ‘001’, the UPID set to ‘Only Idle Data’, and the Last Valid Octet pointer set to the last octet of the TFDZ.

NOTE – This is a complete MAPA_SDU containing Idle Data.

4.1.4.1.10 The Transfer Frame Data Zone of an OID Transfer Frame shall consist of a project-specified ‘idle’ pattern.

NOTES

- 1 OID Transfer Frames are not sent on Virtual Channels that also contain valid Packets.

- 2 OID Transfer Frames are sent to maintain synchronization at the receiver and also to transmit data in the Transfer Frame Insert Zone and/or Operational Control Field when there is neither user data nor protocol data to send in the Transfer Frame Data Zone.
- 3 Idle Data in the Transfer Frame Data Zone of an OID Transfer Frame is not to be confused with the Idle Packet specified in references [12] and [13].
- 5 The idle data pattern used in the OID Transfer Frame is project-specific, but a random pattern is preferred. Problems with the reception of Frames have been encountered because of insufficient randomization.
- 6 The UPID for Only Idle Data is registered in the SANA UPID registry.

4.1.4.1.11 An OID Transfer Frame Data Field shall not contain any valid user or protocol data within it.

4.1.4.1.12 An OID Transfer Frame may contain an Insert Zone (by managed parameter), an OCF, (signaled in Transfer Frame primary header), and an FECF (by managed parameter).

4.1.4.2 Transfer Frame Data Field Header

4.1.4.2.1 General

4.1.4.2.1.1 If present, the TFDF Header shall follow, without gap, the Transfer Frame Primary Header or the Transfer Frame Insert Zone, if present.

4.1.4.2.1.2 The TFDF Header shall consist of 1 to 3 octets and contain the following fields:

- a) Transfer Frame Data Zone (TFDZ) Construction Rules (3 bits, mandatory);
- b) USLP Protocol Identifier (5 bits, mandatory);
- c) First Header/Last Valid Octet Pointer (16 bits, optional).

NOTE – The format of the TFDF Header is shown in figure 4-5.

TFDZ Construction Rules	USLP Protocol Identifier	First Header/ Last Valid Octet Pointer (Optional)
3 bits	5 bits	16 bits

Figure 4-5: Transfer Frame Data Field Header

4.1.4.2.2 TFDZ Construction Rules

4.1.4.2.2.1 General

4.1.4.2.2.1.1 Bits 0–2 of the Transfer Frame Data Field Header shall contain the TFDZ Construction Rules.

4.1.4.2.2.1.2 The TFDZ Construction Rules shall be used to identify how the protocol organizes the user data within the TFDZ in order to transport it.

4.1.4.2.2.1.3 A MAPA_SDU and/or a single Packet SDU may be segmented with portions thereof placed within the TFDZ of successive Frames within the same GMAP ID (VCID + MAP ID).

4.1.4.2.2.1.4 A MAPA_SDU must always begin in the first octet of the TFDZ in the first Frame carrying that MAPA_SDU.

4.1.4.2.2.1.5 The initial Packet placed in a variable-length TFDZ must always begin in the first octet of the TFDZ in the first Frame carrying that Packet.

4.1.4.2.2.1.6 The remaining portions of a Packet or MAPA_SDU placed in a variable-length TFDZ that is longer than the maximum length of the Frame shall be transported in Frames that have the same GMAP ID and have contiguous Virtual Channel Frame Counts.

NOTE – The eight TFDZ Construction Rules are contained in 4.1.4.2.2.2.1 through 4.1.4.2.2.2.8.

4.1.4.2.2.2 Itemized Construction Rules

4.1.4.2.2.2.1 Packets Spanning Multiple Frames. TFDZ Construction Rule ‘000’ shall indicate a fixed-length TFDZ whose contents are CCSDS Packets that are concatenated together and span Transfer Frame boundaries. The First Header Pointer is required for packet extraction.

4.1.4.2.2.2.2 Start of a MAPA_SDU. TFDZ Construction Rule ‘001’ shall indicate a fixed-length TFDZ that contains either a portion of or a complete MAPA_SDU that starts in the first octet of the TFDZ. The Last Valid Octet Pointer shall be set to binary ‘all ones’ if the complete MAPA_SDU is not fully contained within the TFDZ. Otherwise the value in the Last Valid Octet Pointer shall be the delimiter of the MAPA_SDU and point to the last valid octet of the MAPA_SDU.

4.1.4.2.2.2.3 Continuing Portion of a MAPA_SDU. The TFDZ Construction Rule ‘010’ shall indicate a fixed-length TFDZ that contains a portion of a MAPA_SDU that was started in a previous TFDZ. The Last Valid Octet Pointer is required to delimit the end of the MAPA_SDU and shall contain binary ‘all ones’ if the end of the MAPA_SDU is not contained in this TFDZ.

NOTE – The MAPA_SDU begins in the first octet of a TFDZ (Rule ‘001’), and portions of that MAPA_SDU are contained in the TFDZs of the following Transfer Frames of that VC (Rule ‘010’), with the last Frame completing the MAPA_SDU (Rule ‘010’).

4.1.4.2.2.2.4 Octet Stream. The TFDZ Construction Rule ‘011’ shall indicate a variable-length TFDZ containing an octet-aligned stream that is intended to be continuous without beginning or end.

NOTE – An example is video.

4.1.4.2.2.2.5 Starting Segment. The TFDZ Construction Rule ‘100’ shall indicate a variable-length TFDZ containing only the starting segment of an SDU, i.e., either a MAPA_SDU or Packet that is longer than the maximum allowed transmission unit for that GMAP ID.

NOTE – The SDU is segmented and starts but does not end in this TFDZ.

4.1.4.2.2.2.6 Continuing Segment. The TFDZ Construction Rule ‘101’ shall indicate a variable-length TFDZ containing only a continuing segment (i.e., continuing portion of an SDU contained in the previous TFDZ for this GMAP ID).

4.1.4.2.2.2.7 Last Segment. The TFDZ Construction Rule ‘110’ shall indicate a variable-length TFDZ containing only the last segment of the SDU being transferred for this GMAP ID.

4.1.4.2.2.2.8 No Segmentation. The TFDZ Construction Rule ‘111’ shall indicate a variable-length TFDZ that is not segmented, containing either one MAPA_SDU or one or more complete Packets.

NOTES

- 1 A series of complete packets can be placed into the TFDZ of a variable-length Frame if the sum of their individual lengths does not exceed the maximum TFDZ size for the maximum Frame length allowed for that VC and if they have the same quality of service.
- 2 The TFDZ Construction Rules are summarized in table 4-3.

Table 4-3: Summary of the TFDZ Construction Rules

TFDZ Construction Rule Value	Applicable to either Fixed or Variable-Length TFDZs	TFDZ Construction Rule Requirement
000	Fixed	4.1.4.2.2.2.1 Packets Spanning Multiple Frames
001	Fixed	4.1.4.2.2.2.2 Start of MAPA_SDU (Complete or Portion)
010	Fixed	4.1.4.2.2.2.3 Continuing Portion of MAPA_SDU
011	Variable	4.1.4.2.2.2.4 Octet Stream
100	Variable	4.1.4.2.2.2.5 Starting Segment
101	Variable	4.1.4.2.2.2.6 Continuing Segment
110	Variable	4.1.4.2.2.2.7 Last Segment
111	Variable	4.1.4.2.2.2.8 No Segmentation

4.1.4.2.3 USLP Protocol Identifier

4.1.4.2.3.1 Bits 3–7 of the Transfer Frame Data Field Header shall contain the UPID.

4.1.4.2.3.2 The UPID shall identify the CCSDS recognized protocol, procedure, or type of data contained within the TFDZ.

4.1.4.2.3.3 The UPID shall be interpreted as follows:¹

- a) The value ‘00000’ in the UPID field signals that Space Packets or Encapsulation packets are contained within the TFDZ.
- b) The value ‘00001’ in the UPID field signals that COP-1 directives are contained within the TFDZ.
- c) The value ‘00010’ in the UPID field signals that COP-P directives are contained within the TFDZ.
- d) The value ‘00011’ in the UPID field signals that SDLS directives are contained with the TFDZ.

¹ When this Recommended Standard is finalized, the USLP Protocol IDs will be registered and maintained within SANA.

- e) The value '00100' in the UPID field signals that user-defined Octet Stream data are contained within the TFDZ.
- f) The value '00101' in the UPID field signals that mission-specific information-1 as a MAPA_SDU is contained within the TFDZ.
- g) The value '00110' in the UPID field signals that mission-specific information-2 as a MAPA_SDU is contained within the TFDZ.
- h) The value '00111' in the UPID field signals that Proximity-1 Supervisory Protocol Data Units (SPDUs) are contained within the TFDZ.
- i) The value '11111' in the UPID field signals that the entire TFDZ contains only Idle Data.

NOTE – There is no UPID value for either a CLCW or a PLCW, since when present either would reside in the OCF.

4.1.4.2.4 First Header/Last Valid Octet Pointer

4.1.4.2.4.1 If present, the 16-bit First Header/Last Valid Octet Pointer Field shall follow, without gap, the USLP Protocol Identifier field and shall contain the offset to a specific octet within the TFDZ.

4.1.4.2.4.2 The First Header/Last Valid Octet Pointer Field shall be used only for TFDZ Construction Rules '000', '001', and '010'.

NOTE – The first octet in the TFDZ has an offset of '0'.

4.1.4.2.4.3 When the value in the TFDZ construction rule is '000' binary, the First Header/Last Valid Octet field shall contain the offset within the TFDZ to the first octet of the first packet header that starts within the TFDZ. In this case, this field is designated as the First Header Pointer.

NOTE – The function of the First Header Pointer (FHP) is to enable the packet extraction process to restart whenever a Transfer Frame is lost on that VC.

4.1.4.2.4.4 When the value in the TFDZ construction rule is '000' binary, and when no packet starts in the TFDZ, then the FHP shall be set to binary 'all ones'.

NOTES

- 1 The above situation can occur if a long PDU extends across more than one Transfer Frame.
- 2 The purpose of the First Header Pointer is to facilitate delimiting of variable-length packets contained within the TFDZ, by pointing directly to the location of the first packet from which its length may be determined.

- 3 The locations of any subsequent packets within the same TFDZ will be determined by calculating the locations using the length field of these packets.
- 4 If the last packet in the TFDZ of Transfer Frame M spills over into Frame $M+1$ of the same Virtual Channel, then the First Header Pointer in Frame $M+1$ points to the start of the next packet within Frame $M+1$.
- 5 If the last packet in the TFDZ of Transfer Frame M spills over into Frame $M+1$ of the same Virtual Channel but Frame $M+1$ is missing, then the First Header Pointer in the next Frame received in that VC ignores the residue of the split packet and points to the start of the next packet that begins in that Frame.

4.1.4.2.4.5 When the value in the TFDZ Construction Rules is binary ‘001’ or ‘010’, the First Header/Last Valid Octet Pointer field shall contain the offset to the last octet of the MAPA_SDU being transferred, with the remaining octets composed of idle data (a project-specified idle pattern). In this case this field is designated as the Last Valid Octet Pointer.

4.1.4.2.4.6 If the MAPA_SDU does not complete within this fixed-length TFDZ then the value contained within the Last Valid Octet Pointer shall be set to binary ‘all ones’.

NOTE – If the length of the TFDZ is fixed and the MAPA_SDU has an insufficient number of user data octets remaining to complete the TFDZ, then the Pointer field indicates the location of the last valid user data octet within the TFDZ (i.e., the boundary between user data and any inserted idle data).

4.1.4.3 Transfer Frame Data Zone (TFDZ)

4.1.4.3.1 If present, the Transfer Frame Data Zone shall follow, without gap, the TFDZ Header.

4.1.4.3.2 The data contained within any given TFDZ shall be associated with one and only one GMAP ID + QoS + USLP Protocol ID.

4.1.4.3.3 The TFDZ shall contain the data defined by the USLP Protocol ID.

NOTES

- 1 The idle pattern used in the TFDZ is project-specific and can be fixed or variable length. A random pattern is preferred. Problems with the reception of Frames have been encountered because of insufficient randomization.
- 2 Idle data is used only with fixed-length TFDZs.
- 3 When the TFDZ Construction Rule is ‘000’, the first and last packet of the TFDZ are not necessarily complete, since the first packet may be a continuation of a packet begun in the previous TFDZ, and the last packet may continue in the subsequent TFDZ of the same Virtual Channel.

4.1.4.3.4 If a fixed-length TFDZ is partially completed with Packets when the required release time for a Transfer Frame of a Virtual Channel has been reached, an Encapsulation Idle Packet (reference [13]) shall complete the remainder of the TFDZ.

NOTES

- 1 This event can occur based on the timeliness criteria contained in the managed parameters.
- 2 Idle Data in the TFDZ is not to be confused with the Idle Packet specified in reference [12] or [13].

4.1.5 OPERATIONAL CONTROL FIELD

4.1.5.1 If present, the Operational Control Field shall occupy the four octets following, without gap, the TFDF, if this is present, or the Insert Zone, if the TFDF is not present, or the Transfer Frame Primary Header, if the Insert Zone and the TFDF are not present.

4.1.5.2 The Operational Control Field is optional; its presence or absence shall be signaled by the Operational Control Field Flag for each applicable Virtual Channel.

4.1.5.3 Bit 0 of the Operational Control Field shall contain a Type Flag with the following meanings:

- a) Type Flag '0': the Operational Control Field holds a Type-1-Report which contains either a Communications Link Control Word (CLCW) or a Proximity Link Control Word (PLCW);
- b) Type Flag '1': the Operational Control Field holds a Type-2-Report.

NOTE – The value of the Type Flag can vary between Transfer Frames on the same Virtual Channel that carries this field.

4.1.5.4 In a Type-2 Report, bit 1 of the Operational Control Field shall indicate the use of this report as follows:

- a) if this bit is '0', the contents of the report are project-specific;
- b) if this bit is '1', the contents of the report contain an SDLS Frame Security Report (FSR) defined in reference [15].

NOTES

- 1 The Operational Control Field is provided to support the reporting mechanism of the COP-1/COP-P and/or the SDLS protocol.
- 2 In Type-2 Reports, the value of bit 1 of the Operational Control Field may vary between Transfer Frames on the same Virtual Channel that carries this field.

- 3 The purpose of this field is to provide a standardized mechanism for reporting a small number of real-time functions (such as retransmission control or spacecraft clock calibration); currently the use for retransmission control (Type-1 Reports) has been defined by CCSDS in reference [E6].
- 4 If the Master Channel Managed Parameter 'Transfer Frame Type' is 'Fixed Length', and if for at least one VC within the Master Channel, the VC Managed Parameter 'Inclusion of OCF Required' is 'True', then an OCF will always be included in every Frame of that Virtual Channel.
- 5 If the Master Channel Managed Parameter 'Transfer Frame Type' is 'Variable Length' and has subordinate VCIDs with Frame types that are both fixed and variable length, with the associated VC Managed Parameter 'Inclusion of OCF Required' set to 'True' and 'Inclusion of OCF Allowed' set to 'True', then the OCF will be included in every Frame on the fixed-length Frame VCIDs and only as desired based upon mission rules for the variable-length Frame VCIDs.
- 6 The mission could specify for a variable-length Frame that one will only include a new OCF or may require the inclusion of an OCF for a number of times until a new OCF_SDU is received.

4.1.6 FRAME ERROR CONTROL FIELD

4.1.6.1 Overview

The purpose of the Frame Error Control Field is to provide a capability for detecting errors that may have been introduced into the Transfer Frame during the transmission and data handling process.

4.1.6.2 General

4.1.6.2.1 The Frame Error Control Field is optional; its presence or absence shall be established by management based upon the value of the Physical Channel Managed Parameter 'Presence of Frame Error Control'.

4.1.6.2.2 If present, the Frame Error Control Field shall occupy the last 16 or 32 bits of every Transfer Frame based upon the value of the Physical Channel Managed Parameter 'Frame Error Control Length'. If present, the Frame Error Control Field shall occur within every Transfer Frame transmitted within the same Physical Channel throughout a Mission Phase.

NOTES

- 1 Whether this field should be used on a particular Physical Channel is determined based on the mission requirements for data quality and the selected options for the underlying Synchronization and Channel Coding Sublayer. This field may be

mandatory depending on the selected options for the Synchronization and Channel Coding Sublayer as specified in references [3], [6], or [7], or by management for other reasons besides coding.

- 2 CCSDS provides two coding procedure options: 16-bit or 32-bit. The CRC-16 procedure provides an undetected bit error rate of approximately 10^{-5} compared to approximately 10^{-11} for the CRC-32 procedure (see annex B for their definition).

4.2 PROTOCOL PROCEDURES AT THE SENDING END

4.2.1 OVERVIEW

This subsection describes procedures at the sending end associated with each of the functions shown in figure 4-6 (which is identical to figure 2-5). In the figure, data flow from top to bottom. The figure identifies data-handling functions performed by the protocol entity at the sending end, and shows logical relationships among these functions. The figure is not intended to imply any hardware or software configuration in a real system. Depending on the services actually used for a real system, not all the functions may be present in the protocol entity. The procedures described in this subsection are defined in an abstract sense and are not intended to imply any particular implementation approach of a protocol entity.

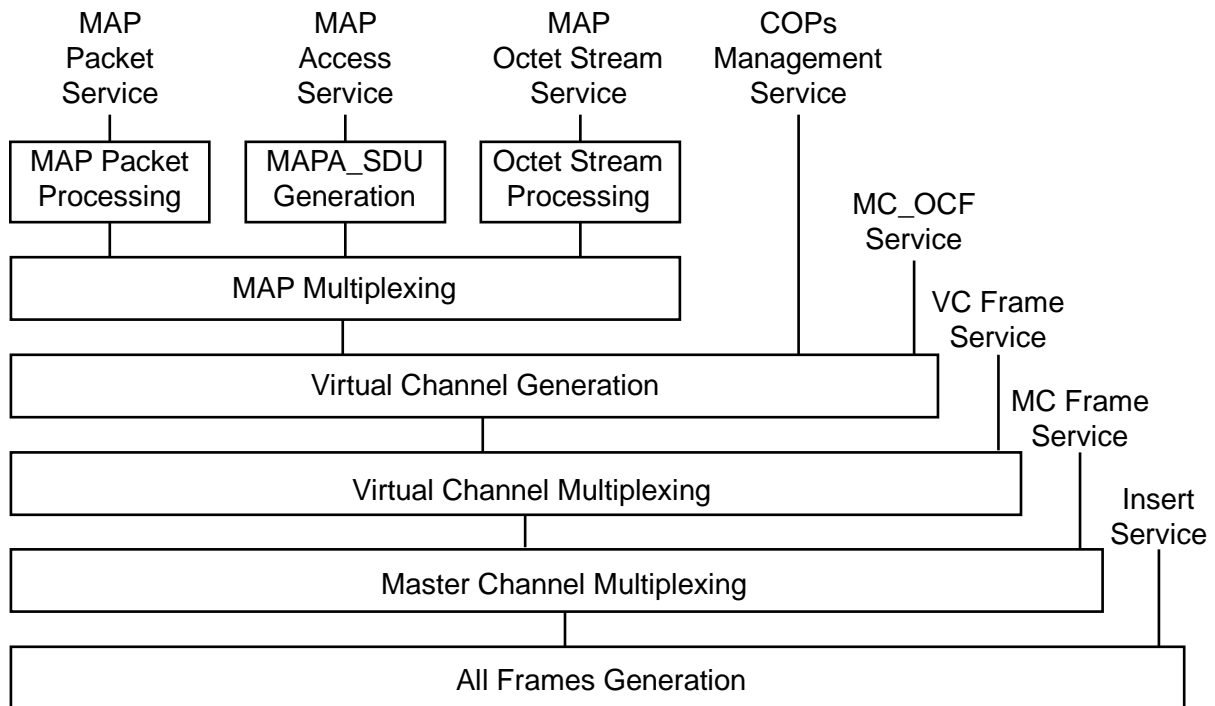


Figure 4-6: Internal Organization of Protocol Entity (Sending End)

4.2.2 MAP PACKET PROCESSING FUNCTION

4.2.2.1 MAP Packet Processing for Fixed-Length TFDZ

4.2.2.1.1 The MAP Packet Processing Function for a fixed-length TFDZ shall be used to transfer variable-length Packets in fixed-length TFDZs of Transfer Frames.

NOTES

- 1 When a packet spans a Transfer Frame, the associated TFDZ Construction Rule used is '000'.
- 2 There is an instance of the Packet Processing Function for each MAP Channel that carries Packets.

4.2.2.1.2 The fixed-length TFDZ that carries Packets shall be constructed by concatenating Packets together until the maximum TFDZ length is exceeded. The Packet whose contents exceed the maximum TFDZ length shall be split, filling the TFDZ completely, and the remainder of the Packet shall be placed in a new TFDZ on the same MAP Channel. Construction of the next and the following TFDZs shall continue with the concatenation of Packets until each TFDZ overflows.

4.2.2.1.3 If Packets of multiple versions are to be transferred on a MAP Channel, Packets of these versions shall be multiplexed into a contiguous string of Packets before constructing TFDZs.

NOTE – The associated TFDZ Construction Rule used is '000'.

4.2.2.1.4 The 'First Header Pointer' field shall be set to indicate the location of the first octet of the first Packet occurring within the TFDZ.

4.2.2.1.5 In the absence of sufficient Packets supplied from the users at release time, one Encapsulation Idle Packet of appropriate length shall be inserted to complete the TFDZ.

NOTES

- 1 An Encapsulation Idle Packet is defined by reference [13]. The shortest Encapsulation Idle Packet defined is one octet in length (i.e., a one-octet header).
- 2 An abstract model of the MAP Packet Processing Function for fixed TFDZs is illustrated in figure 4-7.

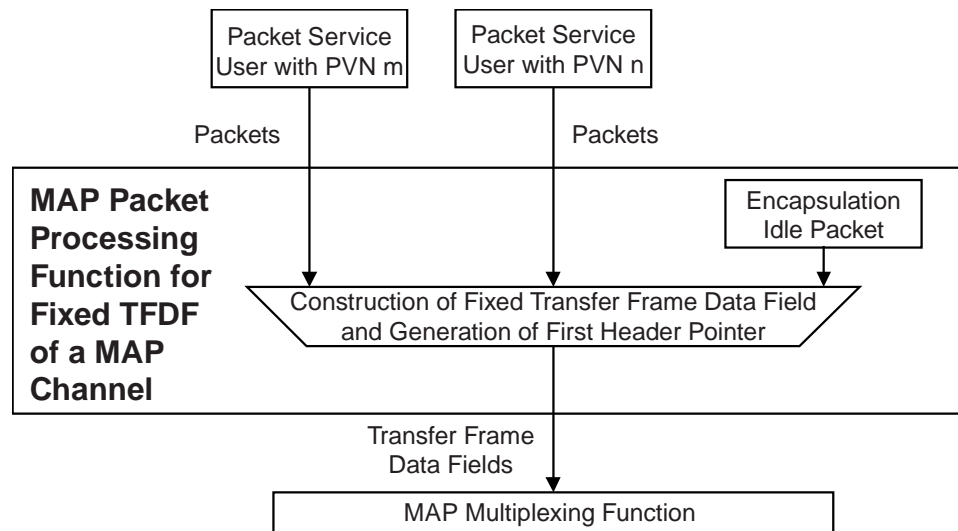


Figure 4-7: Abstract Model of Packet Processing Function for Fixed TFDZs

4.2.2.2 MAP Packet Processing for Variable-Length TFDZ

4.2.2.2.1 The MAP Packet Processing Function for a variable-length TFDZ shall be used to transfer variable-length Packets in the variable-length Data Zone of Transfer Frames of a MAP Channel.

NOTE – There is an instance of the Packet Processing Function for each MAP Channel that carries Packets.

4.2.2.2.2 If the Packet to be transferred exceeds the maximum transmission unit size of the TFDZ, the MAP Packet Processing Function shall divide it into portions that are compatible with insertion into the TFDZ and attach a TFDZ Header to each portion, forming a TFDZ.

4.2.2.2.3 The first octet of the Packet shall appear, without gap, after the TFDZ Header, in the first octet of the TFDZ.

NOTE – The associated TFDZ Construction Rule used is ‘100’.

4.2.2.2.4 The TFDZs containing the first and continuing portions of the Packet may each have a length equal to the maximum allowable length of the TFDZ on that particular MAP Channel.

NOTE – The associated TFDZ Construction Rule used for continuing segments is ‘101’.

4.2.2.2.5 The TFDZ containing the last portion of the Packet shall contain the TFDZ Header and the remainder of the Packet.

NOTE – The associated TFDZ Construction Rule used is ‘110’.

4.2.2.2.6 The portions of a Packet shall be transferred in consecutive Transfer Frames of the MAP Channel without being interlaced with any other Packets or portions in the same MAP Channel.

4.2.2.2.7 If blocking of Packets is permitted on a particular MAP Channel, then:

- a) multiple complete Packets may be placed into a TFDF with a single TFDF Header preceding them;
- b) the blocked Packets plus the TFDF Header must fit within the maximum size TFDF permitted for the Virtual Channel.

NOTES

- 1 The associated TFDZ Construction Rule used is ‘111’.
- 2 An abstract model of the MAP Packet Processing Function for variable TFDFs is illustrated in figure 4-8.

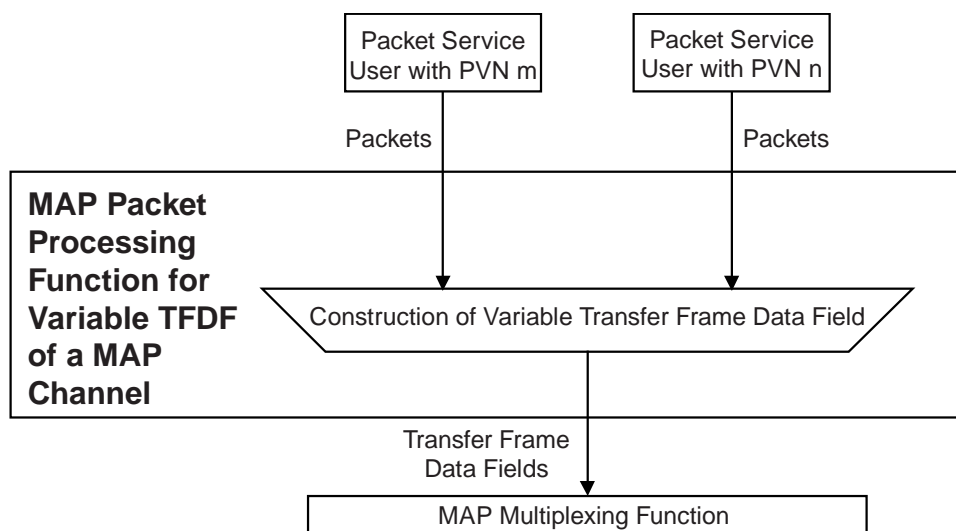


Figure 4-8: Abstract Model of MAP Packet Processing Function for a Variable-Length TFDF

4.2.3 MAPA_SDU GENERATION FUNCTION

4.2.3.1 MAPA_SDU Generation Function for Fixed-Length TFDZ

4.2.3.1.1 The MAPA_SDU Generation Function shall be used to transfer variable-length user-defined service data units (MAPA_SDUs) in fixed-length TFDZs of Transfer Frames of a MAP Channel.

NOTE – There is an instance of the MAPA_SDU Generation Function for each MAP Channel that carries MAPA_SDUs.

4.2.3.1.2 If the MAPA_SDU to be transferred exceeds the maximum transmission unit size of the TFDZ, the MAPA_SDU Generation Function shall divide it into portions that are compatible with insertion into the TFDZ and generate a TFDZ Header consisting of the TFDZ Construction Rule, UPID, and Last Valid Octet Pointer, forming the TFDZs.

NOTE – The TFDZ Construction rule for the first portion is set to ‘001’ while the TFDZ Construction Rule for TFDZs containing the remaining portions will be set to ‘010’ (see 4.1.4.2.2.2).

4.2.3.1.3 If the MAPA_SDU is completely contained in the first TFDZ, then the MAPA_SDU Generation Function shall set the TFDZ Construction Rule to ‘001’ and fill the remainder of the TFDZ with a project-specified Idle Data pattern, and the Last Valid Octet Pointer shall be set to the location of the end of the contained MAPA_SDU.

4.2.3.1.4 If a MAPA_SDU did not complete in a previous TFDZ then the next portion of that same MAPA_SDU shall start in the first octet of the TFDZ of the following Frame in that VC in the same MAP Channel.

NOTE – The associated TFDZ Construction Rule used is ‘010’.

4.2.3.1.5 The Last Valid Octet Pointer shall point to the last octet of the MAPA_SDU if it completes within the TFDZ. If the MAPA_SDU does not complete within the TFDZ then the contents of the Last Valid Octet Pointer shall be set to binary ‘all ones’.

4.2.3.1.6 If the fixed-length TFDZ is not completed with MAPA_SDU data at release time, the MAPA_SDU Generation Function shall fill the remainder of the TFDZ with a project-specified Idle Data pattern.

NOTES

- 1 There can be multiple instances of the MAPA_SDU Generation Function for each Virtual Channel identified by different MAPs. This is possible because a TFDZ that contains a MAPA_SDU can only contain a single MAPA_SDU, thus satisfying the one MAP and one UPID requirement.
- 2 An abstract model of the MAPA_SDU Generation Function for fixed-length TFDZs is illustrated in figure 4-9.

- 3 The boundary between the end of the MAPA_SDU Data and the beginning of the Idle Data is indicated by the Last Valid Octet Pointer.

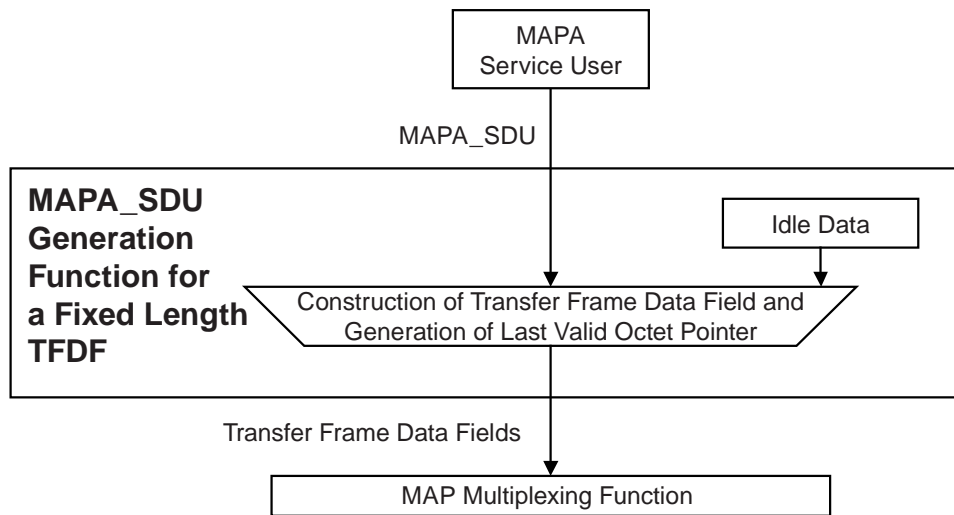


Figure 4-9: Abstract Model of MAPA_SDU Generation Function for Fixed-Length TFDF

4.2.3.2 MAPA_SDU Generation Function for a Variable-Length TFDZ

4.2.3.2.1 The MAPA_SDU Generation Function shall be used to transfer variable-length user-defined service data units (MAPA_SDU) in variable-length Data Zones of Transfer Frames of a MAP Channel.

NOTE – There is an instance of the MAPA_SDU Generation Function for each MAP Channel that carries MAPA_SDU.

4.2.3.2.2 If the MAPA_SDU to be transferred exceeds the maximum transmission unit size of the TFDZ, the MAPA_SDU Generation Function shall divide it into portions that are compatible with insertion into the TFDZ and attach a TFDF Header to each portion, forming a TFDF.

4.2.3.2.3 The first octet of the MAPA_SDU shall appear, without gap, after the TFDF Header, in the first octet of the TFDZ.

NOTE – The associated TFDZ Construction Rule used is ‘100’.

4.2.3.2.4 The TFDFs containing the first and continuing portions of the MAPA_SDU may each have a length equal to the maximum allowable length of the TFDF on that particular MAP Channel.

NOTE – The associated TFDZ Construction Rule used for continuing segments is ‘101’.

4.2.3.2.5 The TFDF containing the last portion of the MAPA_SDU shall contain the TFDF Header and the remainder of the MAPA_SDU.

NOTE – The associated TFDZ Construction Rule used is ‘110’.

4.2.3.2.6 The portions of a MAPA_SDU shall be transferred in consecutive Transfer Frames of the MAP Channel without being interlaced with any other MAPA_SDUs or portions in the same MAP Channel.

NOTE – An abstract model of the MAP Generation Function for variable-length TFDZs is illustrated in figure 4-10.

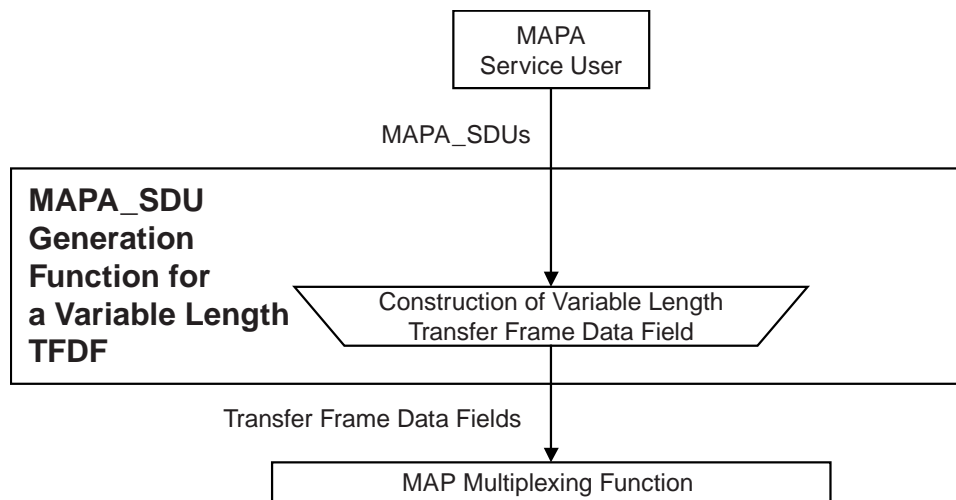


Figure 4-10: Abstract Model of MAPA_SDU Generation Function for Variable-Length TFDF

4.2.4 MAP OCTET STREAM PROCESSING FUNCTION

4.2.4.1 The MAP Octet Stream Processing Function shall be used to transfer variable-length user-defined Octet Stream Data in the variable-length TFDF of Transfer Frames.

NOTE – There is an instance of the Octet Stream Processing Function for each MAP Channel that carries Octet Stream Data.

4.2.4.2 The MAP Octet Stream Processing Function shall be used to complete the TFDZ with the Octet Stream Data supplied by the user.

4.2.4.3 Each octet shall be placed sequentially, and unchanged, into the TFDZ.

4.2.4.4 If the Octet Stream Data have filled one particular TFDZ, the continuation of the Octet Stream Data shall be placed in the next TFDF within the TFDZ on the same MAP Channel.

NOTES

- 1 The associated TFDZ Construction Rule used is ‘011’.
- 2 The release of an Octet Stream TFDF may be subject to release timing constraints contained in the managed parameters.
- 3 An abstract model of the MAP Octet Stream Processing Function is illustrated in figure 4-11.

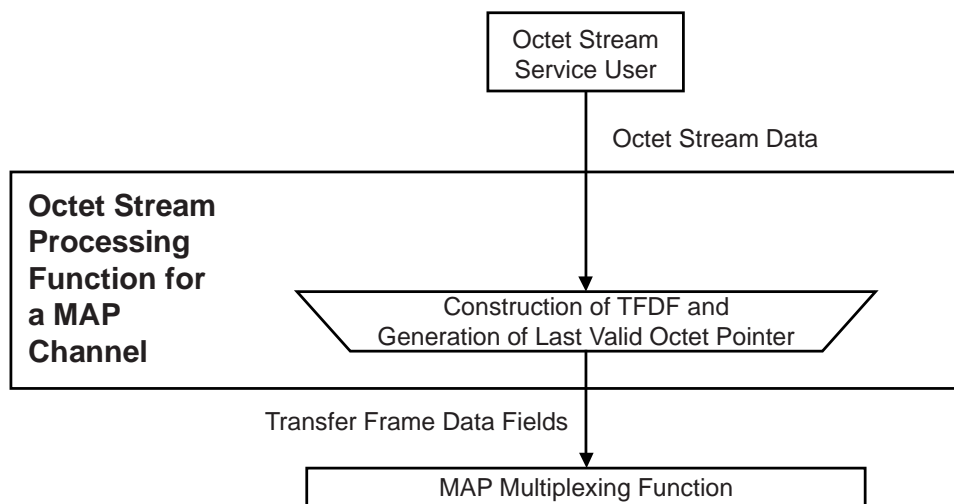


Figure 4-11: Abstract Model of the MAP Octet Stream Processing Function

4.2.5 MAP MULTIPLEXING FUNCTION

4.2.5.1 The MAP Multiplexing Function shall be used to multiplex the contents of the Transfer Frame Data Zone of different MAP Channels of a Virtual Channel.

NOTE – There can be up to 16 instances of the MAP Multiplexing Function for each Virtual Channel.

4.2.5.2 The MAP Multiplexing Function shall multiplex TFDFs as long as their contents are associated with a single MAP ID and a single UPID. These TFDFs shall be put into a queue in an appropriate order set by management.

4.2.5.3 The algorithm to be used to order the service data units is not specified by CCSDS, but shall be defined by project organizations considering factors such as priority, release rate, isochronous timing requirements, etc.

NOTE – An abstract model of the MAP Multiplexing Function is illustrated in figure 4-12.

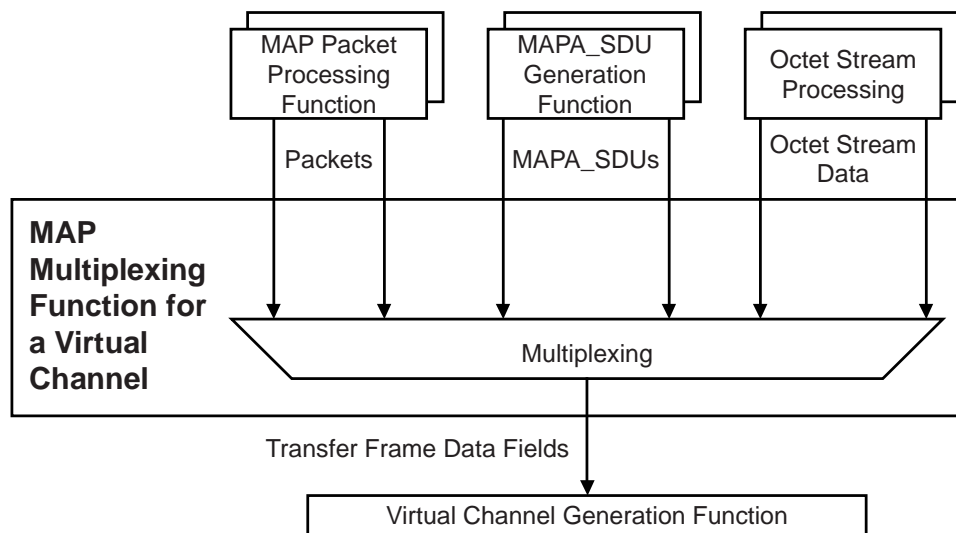


Figure 4-12: Abstract Model of MAP Multiplexing Function

4.2.6 VIRTUAL CHANNEL GENERATION FUNCTION

NOTE – The Virtual Channel Generation Function is used to build the basic structure of Transfer Frames. It is also used to build the structure and the Primary Header of the Transfer Frames for transmission on each Virtual Channel. It also performs most of the operations required to move service data units reliably from the sending end to the receiving end. There is an instance of the Virtual Channel Generation Function for each Virtual Channel.

4.2.6.1 The Virtual Channel Generation Function shall, in the listed order:

- a) maintain, increment, and include the VC Frame Count in the Transfer Frame Primary Header based upon the contents of the Bypass/Sequence Control Flag field;

NOTE – This function also maintains the VC Frame Count Length field.

- b) perform the FOP, which is a sub-procedure of the COP; and

- c) perform the Frame Generation Procedure.

4.2.6.2 The FOP shall accept TFDFs from the MAP Packet Processing Function, Octet Stream Processing Function, or a MAPA Service User, and shall control transmission and retransmission of Frames by examining the report contained in the CLCW/PLCWs and generating Protocol Control Commands.

4.2.6.3 The FOP shall also accept Directives from a COPs Management Service User.

NOTE – The detailed specification of the FOP are given in references [9] and [10].

4.2.6.4 The Frame Generation Procedure shall generate Transfer Frames by attaching a Transfer Frame Primary Header to each TFDF delivered to the FOP.

4.2.6.5 Transfer Frames shall be assembled by placing a single TFDZ unchanged into the Transfer Frame Data Field and generating both the Transfer Frame Primary and TFDF Header fields. A Virtual Channel Frame Count shall be generated independently for each Virtual Channel and placed into the Primary Transfer Frame Header.

4.2.6.6 If there is a user of the MC_OCF Service for a particular Virtual Channel, an OCF_SDU supplied by the user shall be placed in the Operational Control Field.

NOTE – The Virtual Channel Managed Parameters ‘Inclusion of OCF Allowed’ (applicable to variable-length Frames) and ‘Inclusion of OCF Required’ (applicable to fixed-length Frames) determine if an OCF_SDU can be included in that specific Virtual Channel.

4.2.6.7 The Insert Zone and the Frame Error Control Field of Transfer Frames, if present for a particular Physical Channel, shall be kept empty by the Virtual Channel Generation Function.

NOTE – An abstract model of the Virtual Channel Generation Function is illustrated in figure 4-13.

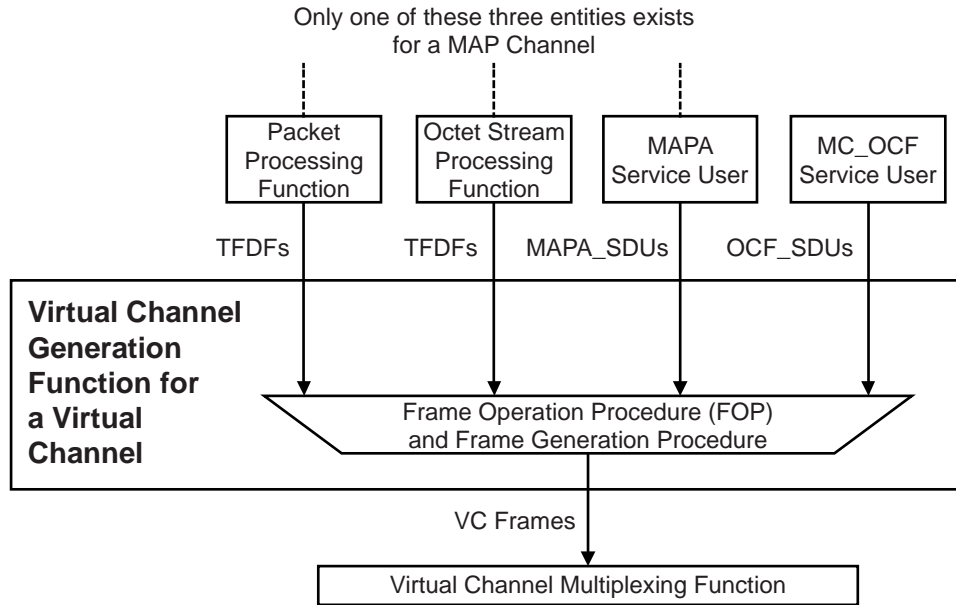


Figure 4-13: Abstract Model of Virtual Channel Generation Function

4.2.7 VIRTUAL CHANNEL MULTIPLEXING FUNCTION

4.2.7.1 The Virtual Channel Multiplexing Function shall be used to multiplex Transfer Frames of different Virtual Channels of a Master Channel.

NOTE – There is an instance of the Virtual Channel Multiplexing Function for each Master Channel that has multiple Virtual Channels.

4.2.7.2 The Virtual Channel Multiplexing Function shall multiplex Transfer Frames received from the instances of the Virtual Channel Generation Function and, if present, the Virtual Channel Frame Service users, in an appropriate order that is set by management.

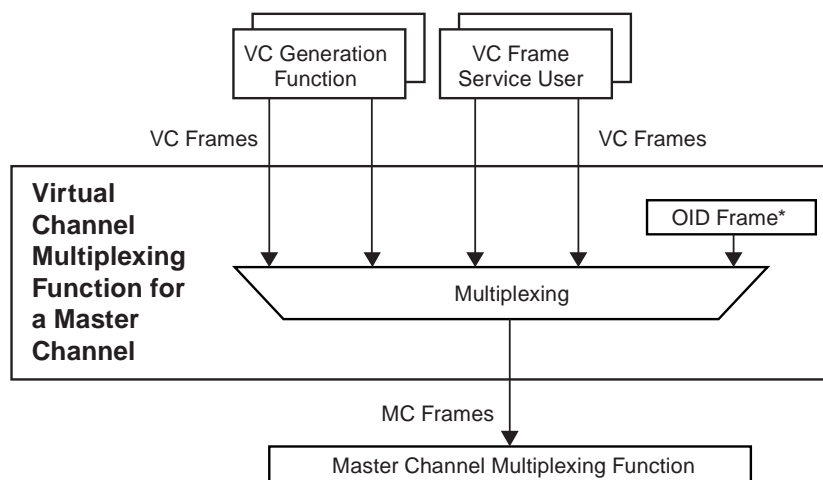
NOTE – The Virtual Channel Multiplexing Function can put the multiplexed Transfer Frames into a queue.

4.2.7.3 The algorithm used to order the Transfer Frames is not specified by CCSDS, but shall be defined by project organizations considering factors such as priority, release rate, isochronous timing requirements, etc.

4.2.7.4 If there is only one Master Channel on the Physical Channel, when the Master Channel Managed Parameter ‘Transfer Frame Type’ equals ‘Fixed length’, the Virtual Channel Multiplexing Function shall create an OID Transfer Frame to preserve the continuity of the transmitted stream in the event that there are no valid Transfer Frames available for transmission at a release time.

NOTE – When the Transfer Frame Type equals ‘Variable Length’, the Virtual Channel Multiplexing Function does not create an OID Transfer Frame.

4.2.7.5 An abstract model of the Virtual Channel Multiplexing Function is illustrated in figure 4-14.



* Applies to fixed-length frames only.

Figure 4-14: Abstract Model of Virtual Channel Multiplexing Function

4.2.8 MASTER CHANNEL MULTIPLEXING FUNCTION

4.2.8.1 The Master Channel Multiplexing Function shall be used to multiplex Transfer Frames of different Master Channels of a Physical Channel.

NOTE – There is an instance of the Master Channel Multiplexing Function for each Physical Channel that has multiple Master Channels.

4.2.8.2 The Master Channel Multiplexing Function shall multiplex Transfer Frames received from the instances of the Virtual Channel Multiplexing Function and, if present, the Master Channel Frame Service users, in an appropriate order that is set by management.

NOTE – The Master Channel Multiplexing Function can put the multiplexed Transfer Frames into a queue.

4.2.8.3 The algorithm to be used to order the Transfer Frames is not specified by CCSDS, but shall be defined by project organizations considering factors such as priority, release rate, isochronous timing requirements, etc.

4.2.8.4 When the Master Channel Managed Parameter ‘Transfer Frame Type’ equals ‘Fixed Length’, the Master Channel Multiplexing Function shall create an OID Transfer Frame to preserve the continuity of the transmitted stream in the event that there are no valid Transfer Frames available for transmission at a release time.

NOTES

- 1 It is not required to maintain a Virtual Channel Frame Count for OID Transfer Frames.
- 2 When the Master Channel (MC) Transfer Frame Type equals ‘Variable Length’, the Master Channel Multiplexing Function does not create an OID Transfer Frame.
- 3 An abstract model of the Master Channel Multiplexing Function is illustrated in figure 4-15.

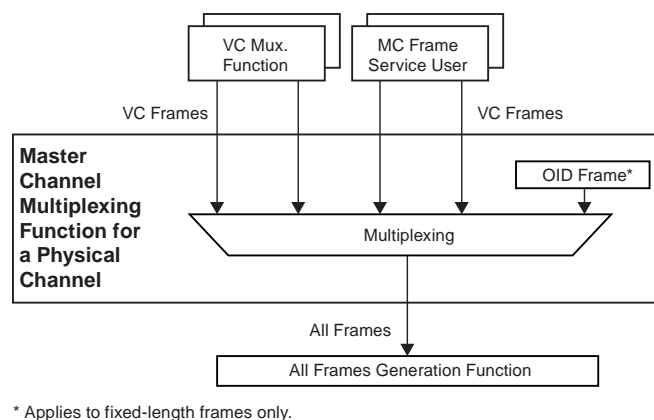


Figure 4-15: Abstract Model of Master Channel Multiplexing Function

4.2.9 ALL FRAMES GENERATION FUNCTION

4.2.9.1 The All Frames Generation Function shall be used to place Insert service data units into Transfer Frames of a Physical Channel. It shall also be used to perform error control encoding defined by this Recommended Standard.

NOTE – There is an instance of the All Frames Generation Function for each Physical Channel.

4.2.9.2 If the optional Insert Service is used, a fixed-length Insert Zone shall exist in every Transfer Frame that is transmitted in a particular Physical Channel. The IN_SDUs shall be timed to arrive at a constant interval that corresponds to the release time of the Transfer Frames onto the Physical Channel. The All Frames Generation Function shall place the IN_SDUs, received from the Insert Service user, into the Insert Zone of the Transfer Frames, preserving octet alignment.

4.2.9.3 If the Frame Error Control Field is present, check bits shall be generated, using the encoding procedure described in annex B, and inserted into the FECF.

4.2.9.4 If the Frame Error Control Field is present, it must be present in all the Transfer Frames transmitted in a particular Physical Channel.

4.2.9.5 Externally generated Transfer Frames associated with the Virtual Channel Frame and Master Channel Frame Services shall always bypass the error control encoding functions specified above.

NOTE – The users of these services therefore need to ensure that the Transfer Frames contain an error control option that conforms with that used by the service provider for the same Physical Channel.

4.2.9.6 The All Frames Generation Function shall deliver data units to the underlying Synchronization and Channel Coding Sublayer.

4.2.9.7 Each data unit delivered by the All Frames Generation Function shall consist of one Transfer Frame.

NOTE – USLP does not use the TC Synchronization and Channel Coding (reference [6]) feature that allows a CLTU to contain multiple Transfer Frames.

4.2.9.8 When reference [6] is used as the Synchronization and Channel Coding Sublayer, the All Frames Generation Function may request the Synchronization and Channel Coding Sublayer to perform systematic retransmissions of a data unit as described in 2.4.2, unless the data unit contains a Frame carrying service data on the Expedited Service.

NOTES

- 1 When systematic retransmissions of a data unit are requested, the additional delay for the retransmissions can be taken into account when deciding the delivery time for the following data unit.
- 2 An abstract model of the All Frames Generation Function is illustrated in figure 4-16.

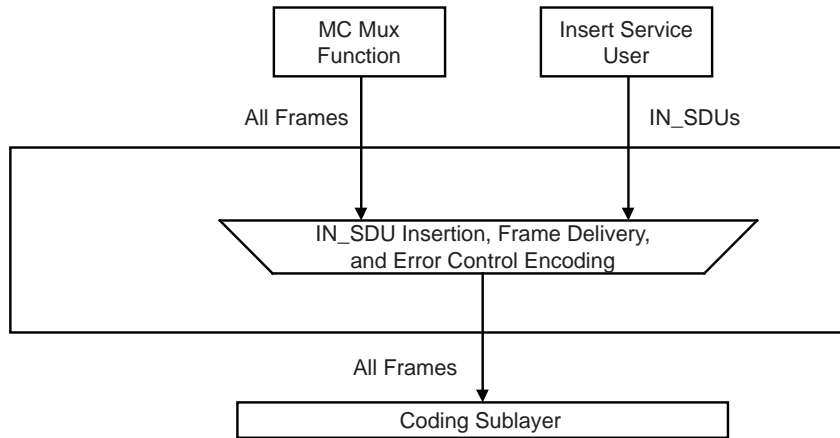


Figure 4-16: Abstract Model of All Frames Generation Function

4.3 PROTOCOL PROCEDURES AT THE RECEIVING END

4.3.1 OVERVIEW

This subsection describes procedures at the receiving end associated with each of the functions shown in figure 4-17 (which is identical to figure 2-6). In the figure, data flows from bottom to top. The figure identifies data-handling functions performed by the protocol entity at the receiving end and shows logical relationships among these functions. The figure is not intended to imply any hardware or software configuration in a real system. Depending on the services actually used for a real system, not all the functions may be present in the protocol entity. The procedures described in this subsection are defined in an abstract sense and are not intended to imply any particular implementation approach of a protocol entity.

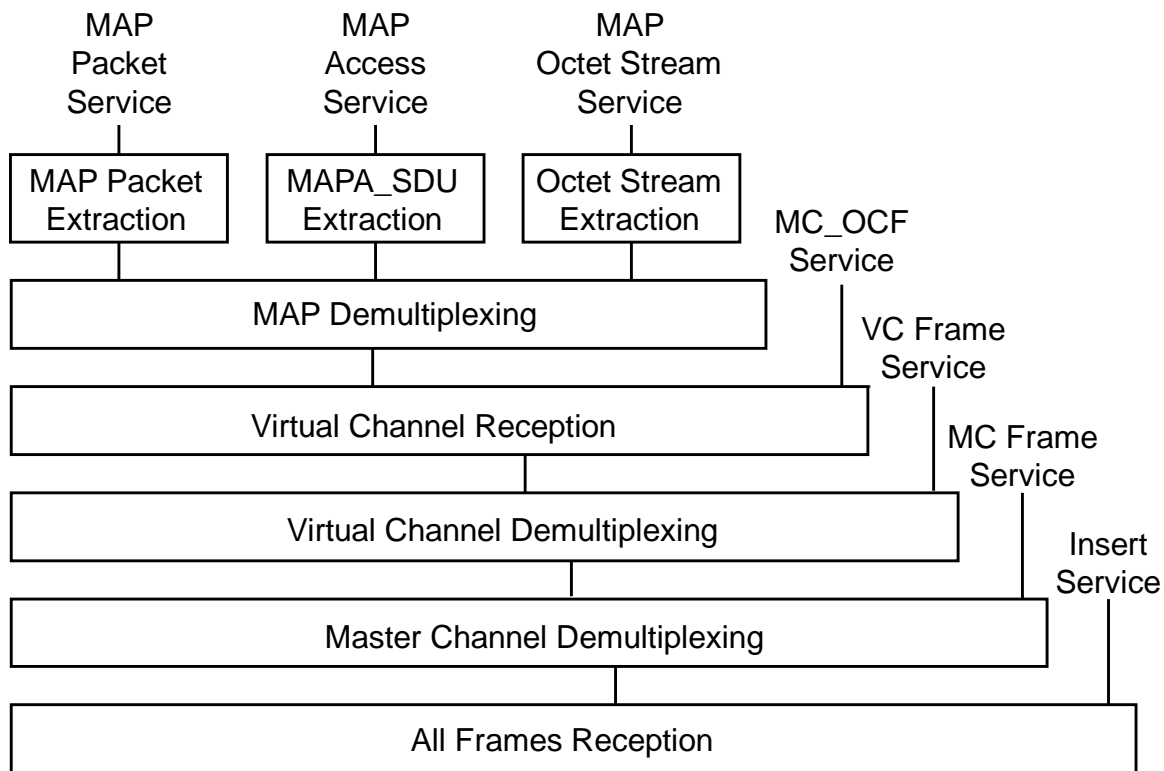


Figure 4-17: Internal Organization of Protocol Entity (Receiving End)

4.3.2 MAP PACKET EXTRACTION FUNCTION

4.3.2.1 Discussion

The extraction of Packets from fixed-length TFDFs is different than from variable-length TFDFs. The TFDZ Construction Rules identify the method used to load the Packets in the TFDF, as well as identify the set of procedures used to extract the packets.

There is an instance of the Packet Extraction Function for each MAP Channel that carries Packets.

4.3.2.2 MAP Packet Extraction Function for a Fixed-Length TFDZ

NOTE – The MAP Packet Extraction Function used to extract variable-length Packets from the fixed-length TFDZs is associated with TFDZ Construction Rule ‘000’ when a packet spans multiple Frames.

4.3.2.2.1 The MAP Packet Extraction Function shall extract Packets from TFDZs received from the MAP Demultiplexing Function.

4.3.2.2.2 The First Header Pointer of each TFDF shall be used in conjunction with the length field of each Packet contained within the TFDZ to provide the delimiting information needed to extract Packets.

4.3.2.2.3 If the last Packet removed from the TFDZ is incomplete, the MAP Packet Extraction Function shall retrieve its remainder from the beginning of the next sequential TFDF received on the same GMAP ID (VCID + MAP ID).

4.3.2.2.4 The First Header Pointer for the next TFDF shall be used to determine the length of the remainder and, hence, the beginning of the next Packet to be extracted.

4.3.2.2.5 If the calculated location of the beginning of the first Packet is not consistent with the location indicated by the First Header Pointer, the MAP Packet Extraction Function shall assume that the First Header Pointer is correct and shall continue the extraction based on that assumption.

NOTES

- 1 Incomplete Packets are not required to be delivered in cross-support situations. Idle Packets are discarded. TFDFs that contain only Idle Data are also discarded.
- 2 An abstract model of the MAP Packet Extraction Function for fixed-length TFDFs is illustrated in figure 4-18.

4.3.2.2.6 MAP Packets associated with an invalid MAP ID shall be discarded.

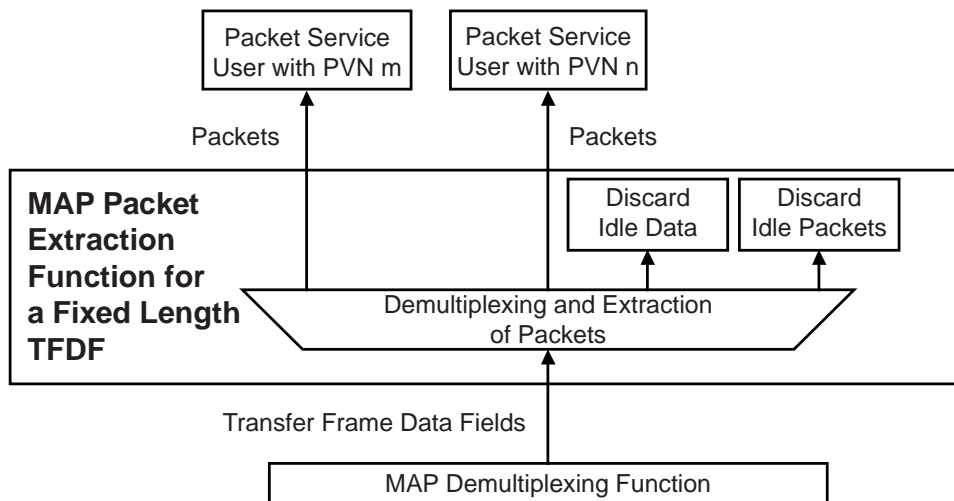


Figure 4-18: Abstract Model of MAP Packet Extraction Function for Fixed-Length TFDFs

4.3.2.3 MAP Packet Extraction Function for a Variable-Length TFDZ

NOTE – The MAP Packet Extraction Function used to extract variable-length Packets from variable-length TFDFs on a MAP Channel is associated with Construction Rules ‘100’, ‘101’, ‘110’ and ‘111’.

4.3.2.3.1 The MAP Packet Extraction Function shall extract Packets from variable-length TFDFs received from the MAP Demultiplexing Function.

4.3.2.3.2 A Segmented Packet within a variable-length TFDF shall be extracted and reconstructed from the TFDF using the TFDZ Construction Rules ‘100’, ‘101’, and ‘110’.

4.3.2.3.3 If blocking of Packets is permitted within a variable-length TFDF, these packets or one complete packet shall be extracted and reconstructed from the TFDF using the TFDZ Construction Rule ‘111’ and the length field of each Packet.

NOTE – Incomplete Packets are not required to be delivered in cross-support situations.

4.3.2.3.4 MAP Packets associated with an invalid MAP ID shall be discarded.

NOTE – An abstract model of the MAP Packet Extraction Function for a variable-length TFDZ is illustrated in figure 4-19.

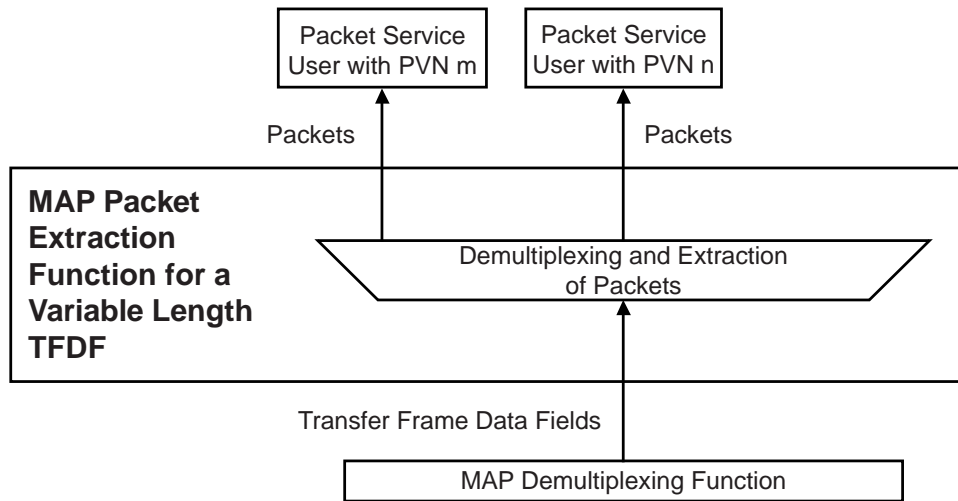


Figure 4-19: Abstract Model of MAP Packet Extraction Function for Variable-Length TFDFs

4.3.3 MAPA_SDU EXTRACTION FUNCTION

4.3.3.1 Overview

The MAPA_SDU Extraction Function extracts one and only one variable-length MAPA_SDU from TFDs delivered by the MAP Demultiplexing Function and provides it to the MAPA Service user. A different set of TFDZ Construction rules is used to extract a MAPA_SDU from a fixed-length TFD than from a variable-length TFD.

NOTE – There is an instance of the MAP Reception Function for each MAP Channel that carries a MAPA_SDU.

4.3.3.2 MAPA_SDU Extraction Function for Fixed-Length TFDs

4.3.3.2.1 The MAPA_SDU Extraction Function shall extract the MAPA_SDU from fixed-length TFDs received from the MAP Demultiplexing Function based upon the GMAP_ID.

4.3.3.2.2 Any Idle Data inserted by the MAPA_SDU Generation Function shall be removed and discarded prior to delivery to the MAPA Service user, using the value in the Last Valid Octet Pointer.

4.3.3.2.3 The MAPA_SDU shall be extracted and reconstructed from TFDs using the TFDZ Construction Rules ‘001’ and ‘010’ and the Last Valid Octet Pointer within each of the TFD Headers.

4.3.3.2.4 The Last Valid Octet Pointer that contains a value that is equal to neither ‘all ones’ nor ‘all ones minus one’ delimits the end of the MAPA_SDU within the TFDZ, and any octets that are beyond that pointer value till the end of the TFDZ shall be discarded.

4.3.3.2.5 The extracted and reconstructed MAPA_SDU shall be delivered to the MAPA Service user associated with its VCID and MAP ID.

4.3.3.2.6 MAPA_SDUs associated with an invalid MAP ID shall be discarded.

NOTE – An abstract model of the MAPA_SDU Extraction Function for fixed-length TFDs is illustrated in figure 4-20.

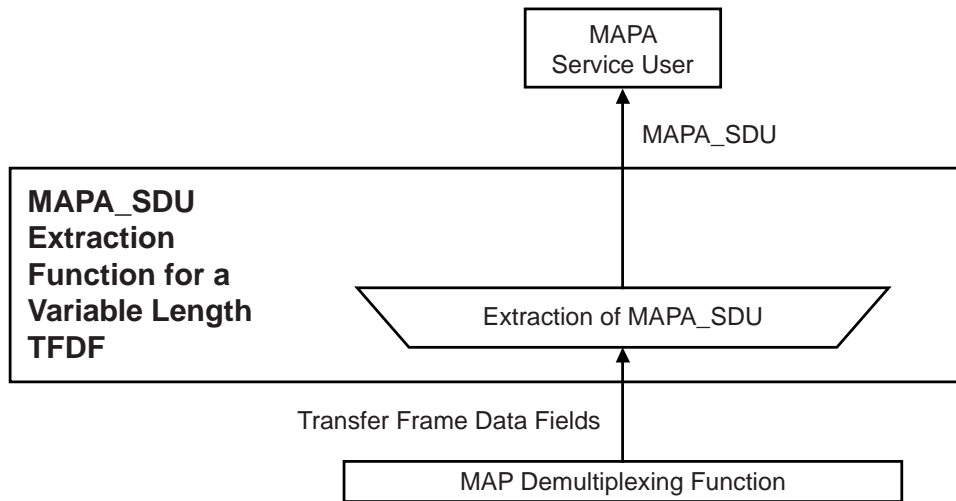


Figure 4-21: Abstract Model of MAPA_SDU Extraction Function for Variable-Length TFDs

4.3.4 MAP OCTET STREAM EXTRACTION FUNCTION

4.3.4.1 The Octet Stream Extraction Function shall be used to extract variable-length user defined Octet Stream Data from variable-length TFDZs on a MAP Channel and deliver the octet stream to the user.

NOTE – There is an instance of the Octet Stream Extraction Function for each MAP Channel that carries Octet Stream Data.

4.3.4.2 The Octet Stream Extraction Function shall extract user-defined Octet Stream Data from TFDZs received from the MAP Demultiplexing Function. The TFDZ Construction Rule used shall be ‘011’.

4.3.4.3 Extracted Octet Stream Data shall be delivered to the MAP Octet Stream Service user identified by the MAP ID.

4.3.4.4 Octet Stream Data associated with an invalid MAP ID shall be discarded.

NOTE – An abstract model of the MAP Octet Stream Extraction Function is illustrated in figure 4-22.

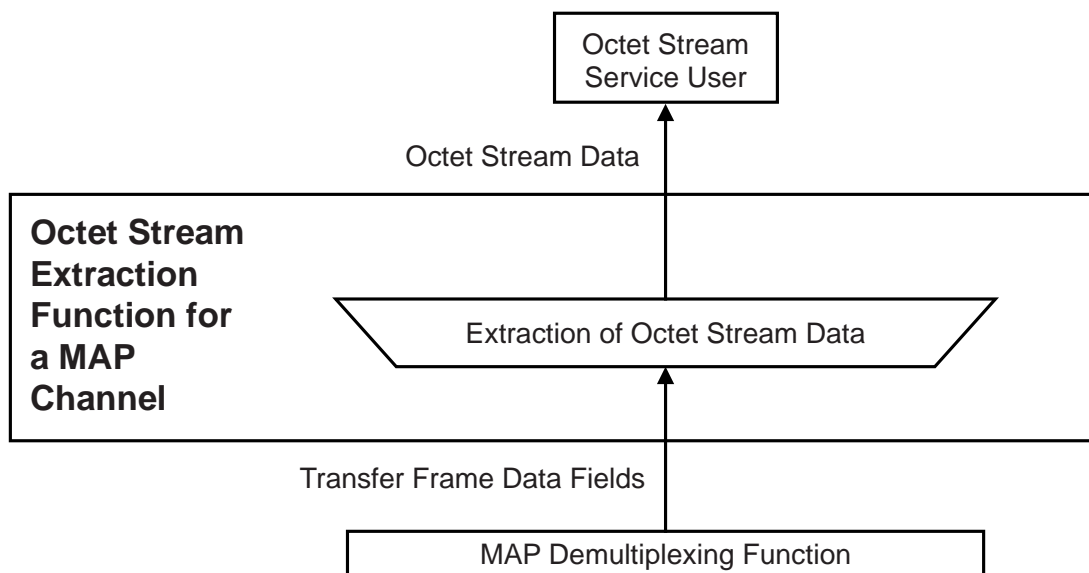


Figure 4-22: Abstract Model of MAP Octet Stream Extraction Function

4.3.5 MAP DEMULTIPLEXING FUNCTION

4.3.5.1 The MAP Demultiplexing Function shall be used to demultiplex service data units from different MAP Channels on a Virtual Channel.

NOTE – There is an instance of the MAP Demultiplexing Function for each Virtual Channel that has multiple MAP Channels.

4.3.5.2 The MAP Demultiplexing Function shall examine the MAP ID in the Transfer Frame Header of the incoming service data units and shall route them to the instances of the Packet Extraction, MAPA_SDU Extraction, and/or Octet Stream Extraction Functions.

4.3.5.3 Service data units associated with an invalid MAP ID shall be discarded.

NOTE – An abstract model of the MAP Demultiplexing Function is illustrated in figure 4-23.

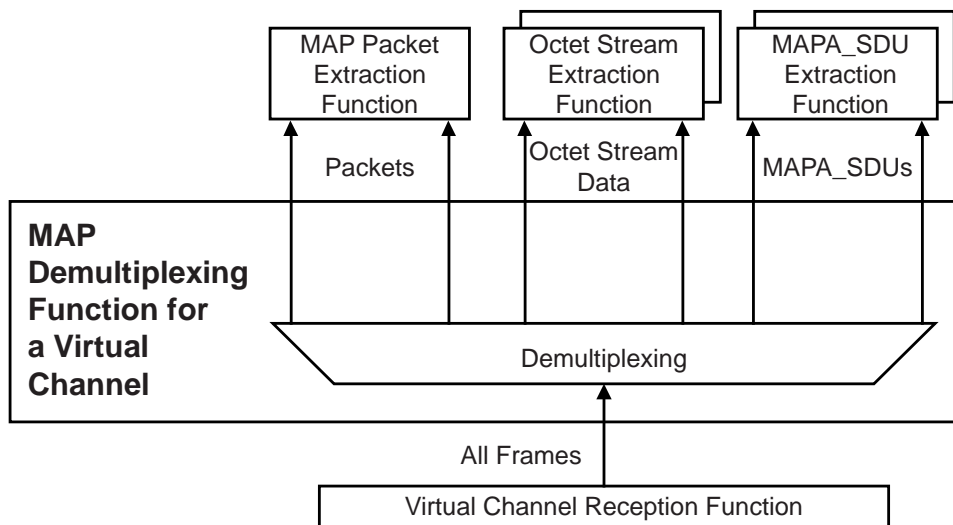


Figure 4-23: Abstract Model of MAP Demultiplexing Function

4.3.6 VIRTUAL CHANNEL RECEPTION FUNCTION

4.3.6.1 The Virtual Channel Reception Function shall be used to decommutate fields of Transfer Frames of a Virtual Channel.

NOTE – There is an instance of the Virtual Channel Reception Function for each Virtual Channel.

4.3.6.2 The Virtual Channel Reception Function shall extract data units contained in the Transfer Frame Data Field and deliver them to the user (i.e., the Packet Extraction Function, the Octet Stream Extraction Function, or the MAPA_SDU Extraction Function).

4.3.6.3 If there is a user of the MC_OCF Service for a particular Virtual Channel, OCF_SDUs contained in the Operational Control Field of the Transfer Frames shall be extracted and delivered to the user.

4.3.6.4 If a gap in the Virtual Channel Frame Count is detected, a Loss Flag may (optionally) be delivered to the MAP receivers associated with this virtual channel.

NOTE – An abstract model of the Virtual Channel Reception Function is illustrated in figure 4-24.

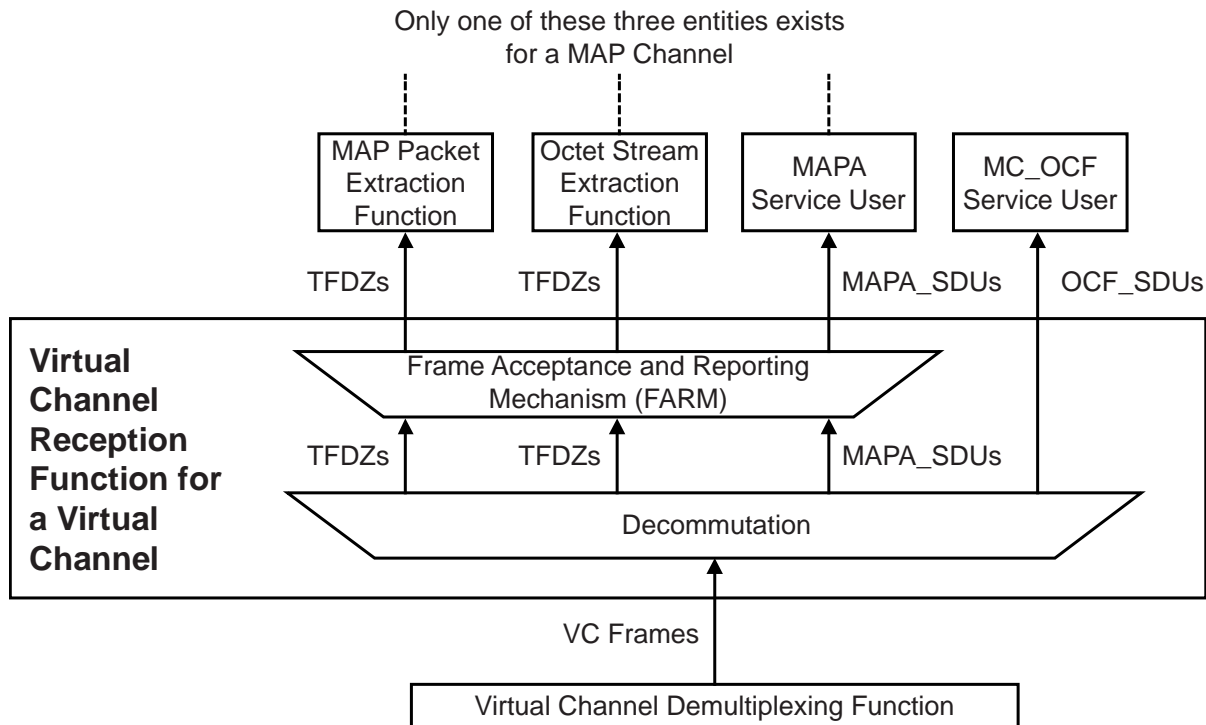


Figure 4-24: Abstract Model of Virtual Channel Reception Function

4.3.7 VIRTUAL CHANNEL DEMULTIPLEXING FUNCTION

4.3.7.1 The Virtual Channel Demultiplexing Function shall be used to demultiplex Transfer Frames of different Virtual Channels of a Master Channel.

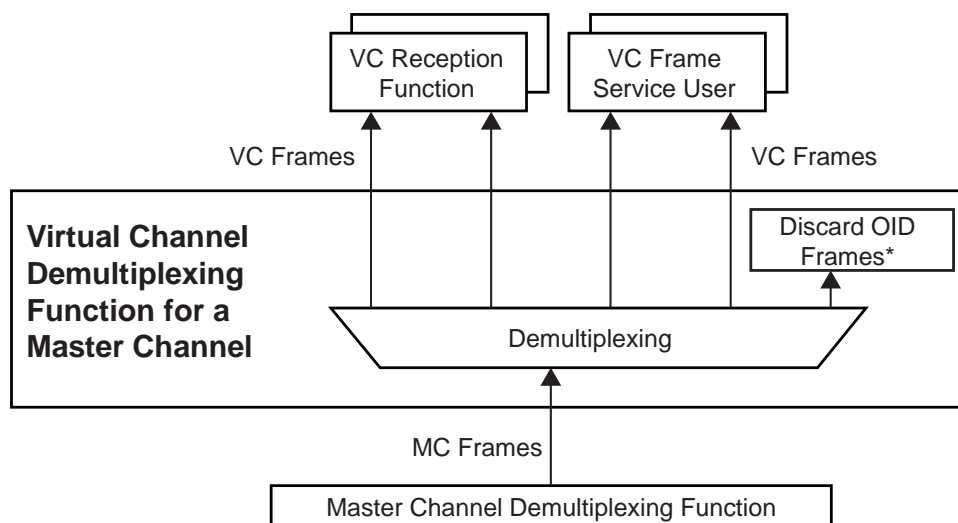
NOTE – There is an instance of the Virtual Channel Demultiplexing Function for each Master Channel that has multiple Virtual Channels.

4.3.7.2 The Virtual Channel Demultiplexing Function shall examine the VCID in the incoming stream of Transfer Frames and route them to the instances of the Virtual Channel Reception Function and, if present, to the Virtual Channel Frame Service users.

4.3.7.3 If a gap in the Virtual Channel Frame Count is detected, a Loss Flag may (optionally) be delivered to the users.

NOTES

- 1 Any OID Transfer Frames encountered are discarded.
- 2 Transfer Frames with an invalid VCID are also discarded.
- 3 An abstract model of the Virtual Channel Demultiplexing Function is illustrated in figure 4-25.



* Applies to fixed-length frames only.

Figure 4-25: Abstract Model of Virtual Channel Demultiplexing Function

4.3.8 MASTER CHANNEL DEMULTIPLEXING FUNCTION

4.3.8.1 The Master Channel Demultiplexing Function shall be used to demultiplex Transfer Frames of different Master Channels of a Physical Channel.

NOTE – There is an instance of the Master Channel Demultiplexing Function for each Physical Channel that has multiple Master Channels.

4.3.8.2 The Master Channel Demultiplexing Function shall examine the MCID in the incoming stream of Transfer Frames and route them to the instances of the Virtual Channel Demultiplexing Function and, if present, to the Master Channel Frame Service users.

4.3.8.3 If Frame loss is signaled by the underlying Synchronization and Channel Coding Sublayer, a Loss Flag may (optionally) be delivered to the users.

4.3.8.4 Transfer Frames with an invalid MCID shall be discarded.

NOTE – An abstract model of the Master Channel Demultiplexing Function is illustrated in figure 4-26.

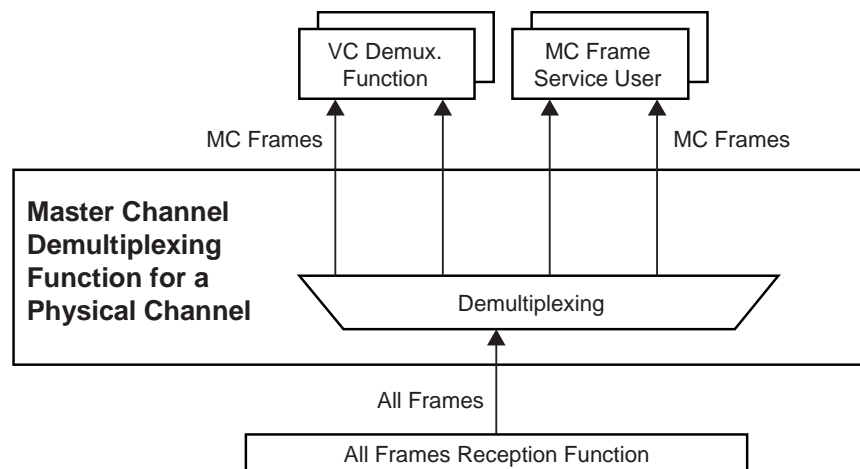


Figure 4-26: Abstract Model of Master Channel Demultiplexing Function

4.3.9 ALL FRAMES RECEPTION FUNCTION

4.3.9.1 General

NOTE – There is an instance of the All Frames Reception Function for each Physical Channel.

4.3.9.1.1 If the Synchronization and Channel Coding Sublayer in use is *TM Synchronization and Channel Coding* (reference [3]), *Flexible Advanced Coding and Modulation Scheme for High Rate Telemetry Applications* (reference [4]), *CCSDS Space Link Protocols over ETSI DVB-S2 Standard* (reference [5]), or *Proximity-1 Space Link Protocol—Coding and Synchronization Sublayer* (reference [7]), the All Frames Reception Function may optionally discard a Transfer Frame that has been validated by the Synchronization and Channel Coding Sublayer if the Synchronization and Channel Coding Sublayer indicates that the Frame was received with error.

NOTE – The Synchronization and Channel Coding Sublayers defined in references [3], [4], [5], and [7] deliver validated Transfer Frames and can optionally signal whether individual Frames were received error free or with error.

4.3.9.1.2 If the Frame Error Control Field is present in the Transfer Frame, the All Frames Reception Function shall recompute the CRC value for the Transfer Frame and compare it to the content of the Frame Error Control field to determine if the Transfer Frame contains a detected error.

4.3.9.1.3 A Transfer Frame that contains a detected error may be discarded in cross-support situations.

4.3.9.1.4 If the optional Insert Service is used, the All Frames Reception Function shall extract the IN_SDU from the Insert Zone of the incoming stream of Transfer Frames, whose Physical Channel Managed Parameter ‘Transfer Frame Type’ is ‘Fixed Length’, regardless of their GMAP ID, and deliver them to the Insert Service user.

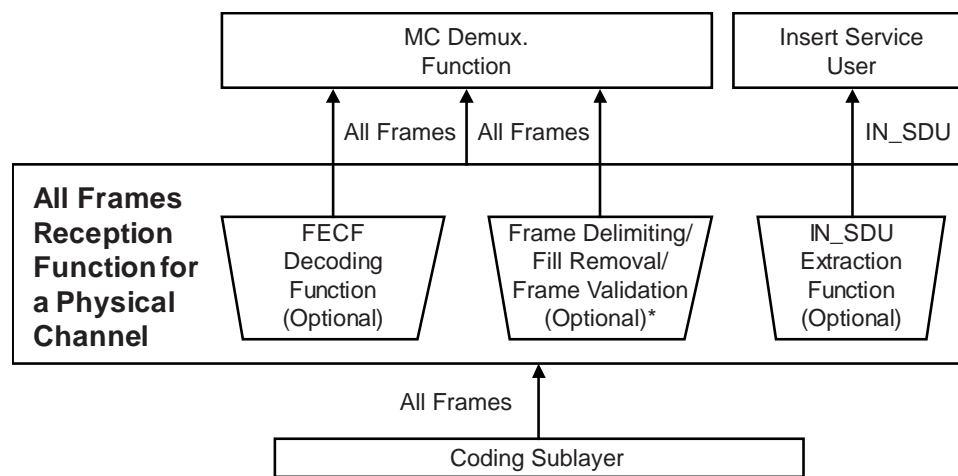
4.3.9.1.5 If error protection of the IN_SDU is not required, extraction of the IN_SDU may be performed prior to decoding of Frame Error Control Field described above.

4.3.9.1.6 If the Synchronization and Channel Coding Sublayer used is *TC Synchronization and Channel Coding* (reference [6]), then the All Frames Reception Function shall be used to reconstitute a single Transfer Frame from the error free data stream provided by the TC Channel Coding Sublayer and perform the following two procedures to determine whether the reconstituted Transfer Frame is valid or not in the following order:

- a) Frame Delimiting and Fill Data Removal Procedure (4.3.9.2); and
- b) USLP Frame Validation Check Procedure (4.3.9.3).

NOTES

- 1 The Frame Delimiting and Fill Data Removal Procedure is used to reconstitute Transfer Frames from the data stream provided by the TC Synchronization and Channel Coding Sublayer and to remove any Fill Data transferred from the TC Synchronization and Channel Coding Sublayer.
- 2 The USLP Frame Validation Check Procedure is used to perform Frame Validation Checks on all Transfer Frames reconstituted by the Frame Delimiting and Fill Data Removal Procedure.
- 3 An abstract model of the All Frames Reception Function is illustrated in figure 4-27.



*Applicable only when TC Synchronization and Channel Coding (reference [6]) is used.

Figure 4-27: Abstract Model of All Frames Reception Function

4.3.9.2 Frame Delimiting and Fill Data Removal Procedure

NOTES

- 1 At the sending end, USLP passes one Transfer Frame to the TC Synchronization and Channel Coding Sublayer at a time.
- 2 The TC Synchronization and Channel Coding Sublayer encodes the Transfer Frame to protect it from errors that may be introduced as it is transmitted through the space link. Fill Data may have to be inserted by the TC Synchronization and Channel Coding Sublayer to ensure correct transmission of all valid data.

4.3.9.2.1 The receiving end of this protocol shall receive as an input from the TC Synchronization and Channel Coding Sublayer a series of error-free data octets, corresponding to the decoded Transfer Frame, possibly incomplete or containing fill data.

NOTE – The TC Synchronization and Channel Coding Sublayer provides a ‘Data Start’ signal to this protocol, indicating that data are being transferred. The Data Start signal is set to ‘true’ while the TC Synchronization and Channel Coding Sublayer is in the process of actively transferring data octets. Since the first octet transferred after Data Start goes ‘true’ corresponds to the first octet of the Transfer Frame, this Procedure may delimit this Transfer Frame by reading the Frame Length field in the first Transfer Frame Header. The Data Start signal is set to ‘false’ (indicating ‘Data Stop’) when the TC Synchronization and Channel Coding Sublayer stops transferring octets because of a decoder failure or channel deactivation. Decoding failure may be caused by the normal end of the transmitted Transfer Frame or by a genuine channel-induced error.

4.3.9.2.2 If one valid Frame Length field is detected by the Frame Delimiting and Fill Data Removal Procedure, and the number of octets received when the Data Stop condition occurs equals the number of octets specified by the Frame Length, then the Transfer Frame shall be passed on to the USLP Frame Validation Check Procedure (see 4.3.9.3) as it is delimited.

4.3.9.2.3 If a valid Frame Length field is detected by the Frame Delimiting and Fill Data Removal Procedure but the number of octets received when the Data Stop condition occurs is fewer than the number of octets specified by that Frame Length, then all those octets shall be discarded.

NOTE – Receipt of fewer octets than specified in Frame Length field indicates that a failure has occurred, possibly resulting from a channel error detected during reception of the data stream within the TC Synchronization and Channel Coding Sublayer.

4.3.9.2.4 If a valid Frame Length field is detected by the Frame Delimiting and Fill Data Removal Procedure but the number of octets received when the Data Stop condition occurs is greater than the number of octets specified by that Frame Length, the procedure shall

- a) assume that the octets following the final expected octet of the Frame are Fill Data appended by the sending end of the Synchronization and Channel Coding Sublayer to complete the last codeword (see reference [6]);
- b) discard that Fill Data;
- c) pass the Frame to the USLP Frame Validation Check Procedure (see 4.3.9.3).

NOTES

- 1 Because the receiving end of the TC Synchronization and Channel Coding Sublayer cannot distinguish between valid data and Fill Data, the Fill Data needs to be stripped by this protocol.

- 2 If fewer than five trailing octets of Fill Data are present, then they cannot possibly form a Transfer Frame Header, and they will be immediately discarded by this procedure. If five or more trailing octets of Fill Data exist (up to six are possible for the BCH code, up to seven for the (128,64) LDPC code, and up to 31 for the (512,256) LDPC code), this procedure might attempt to interpret the Fill Data as a new Transfer Frame Header. In most cases, the subsequent Frame Validation Checks (see 4.3.9.3) will prevent this from happening because the Fill pattern of '01010101' appearing in each octet will violate at least one of the validation tests; in particular, this pattern appearing where the Frame Length field might be expected will indicate a Frame length that exceeds the number of octets received from the Channel Coding Sublayer. The only exception is with the BCH code, if randomization is used, and if the fill data is added after randomization at the transmitting side. In this case, the fill octets will be derandomized without having been randomized, and there is no guarantee that the result will fail the Frame Validation Checks.
- 3 After each Transfer Frame is reconstituted by the Frame Delimiting and Fill Data Removal Procedure, it will next be subjected to a set of tests called USLP Frame Validation Checks.

4.3.9.3 USLP Frame Validation Check Procedure

4.3.9.3.1 The USLP Frame Validation Check Procedure shall be used if *TC Synchronization and Channel Coding* (reference [6]) is the specification used for the Synchronization and Channel Coding Sublayer immediately below this procedure.

4.3.9.3.2 The USLP Frame Validation Checks shall be applied to all incoming Transfer Frames.

4.3.9.3.3 Failure to pass any test within these Checks shall cause the Transfer Frame to be rejected (discarded).

4.3.9.3.4 The USLP Frame Validation Checks shall consist of the following tests:

- a) The Transfer Frame must have an expected Transfer Frame Version Number.
- b) The Transfer Frame must have one of the expected MCIDs (Transfer Frame Version Number and Spacecraft IDs).
- c) The Transfer Frame Header must not contain any values that are not consistent with the implemented features for that spacecraft.
- d) The value of the Frame Length must be consistent with the number of octets that are present.
- e) If the Frame Error Control Field is present, the recomputed CRC value for the Transfer Frame must match the content of the Frame Error Control Field.

5 MANAGED PARAMETERS WITHOUT SDLS OPTION

5.1 OVERVIEW

In this section, the managed parameters used by USLP are listed for each of the Channels and for Packet transfer. These parameters are defined in an abstract sense and are not intended to imply any particular implementation of a management system.

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 protocol entities associated with a particular mission. Through the use of a management system, management conveys the required information to the protocol entities.

NOTE – This section specifies managed parameters for USLP without support for the SDLS protocol. Additional managed parameters for USLP with the SDLS option are specified in 6.6.

5.2 MANAGED PARAMETERS FOR A PHYSICAL CHANNEL

The managed parameters associated with a Physical Channel shall conform to the definitions in table 5-1.

Table 5-1: Managed Parameters for a Physical Channel

Managed Parameter	Allowed Values
Physical Channel Name	Character String
Transfer Frame Type	Fixed Length or Variable Length
Transfer Frame Length (octets)	Integer (maximum 65536 octets)
Transfer Frame Version Number	'1100' binary
MC Multiplexing Scheme	Mission Specific
Presence of Insert Zone	Present ('1'), Absent ('0')
Insert Zone Length (octets)	Integer
Presence of Frame Error Control	Present ('1'), Absent ('0')
Frame Error Control Length (octets)	2 or 4
Generate OID Frame	True ('1'), False ('0')
Maximum Number of Transfer Frames Given to the Coding and Synchronization Sublayer as a Single Data Unit. NOTE – This constraint applies even if reference [6] is used.	1
Maximum Value of the Repetitions Parameter to the Coding and Synchronization Sublayer	Integer
<p>NOTES</p> <p>1 The Transfer Frame Type shall be fixed-length when USLP is used over references [3] and [5]. It can be variable-length when USLP is used over references [6] and [10].</p> <p>2 The value of the Transfer Frame Length shows either the Frame length (when Transfer Frame Type is fixed-length) or the maximum allowed Frame length (when Transfer Frame Type is variable-length).</p> <p>3 The number of allowed values of the Transfer Frame Length may be constrained by the actual coding scheme selected when USLP is used over references [3], [4], and [5].</p> <p>4 The Maximum Number of Transfer Frames given to the Coding and Synchronization Sublayer applies only when USLP is used over reference [6].</p> <p>5 OID Frames are only allowed to be generated for fixed length Transfer Frames.</p>	

5.3 MANAGED PARAMETERS FOR A MASTER CHANNEL

The managed parameters associated with a Master Channel shall conform to the definitions in table 5-2.

Table 5-2: Managed Parameters for a Master Channel

Managed Parameter	Allowed Values
Transfer Frame Type	Fixed Length or Variable Length
Spacecraft ID	16-bit Integer
VCIDs	Selectable Set of Integers (from 0 to 62 with 63 reserved)
VC Multiplexing Scheme	Mission Specific
<p>NOTES</p> <ol style="list-style-type: none"> 1 The value of the Transfer Frame length shows either the Frame length (when Transfer Frame length is fixed-length) or the maximum allowed Frame length (when Transfer Frame length is variable-length). 2 The value of the Transfer Frame Version Number is the same for all Transfer Frames on a Master Channel. 3 For VCID the binary value of 'all ones' (i.e., 63) is always valid as it is reserved for OID Transfer Frames by 4.1.4.1.5; i.e., the number of valid VCIDs always includes value 63 and the Selectable Set of Integers defined above. 4 Transfer Frame Type must be fixed, if the Physical Channel Transfer Frame Type is fixed. 	

5.4 MANAGED PARAMETERS FOR A VIRTUAL CHANNEL

The managed parameters associated with each valid Virtual Channel shall conform to the definitions in table 5-3.

Table 5-3: Managed Parameters for a Virtual Channel

Managed Parameter	Allowed Values
Transfer Frame Type	Fixed Length or Variable Length
VCID	One value from a selectable set of integers (from 0 to 62, with 63 reserved)
VC Count Length for Sequence Control	Integer (maximum 56-bit)
VC Count Length for Expedited Integer	Integer (maximum 56-bit)
COP in Effect	COP-1, COP-P, None
CLCW Version Number	1
CLCW Reporting Rate	as required to support COP
MAP IDs	Selectable set of integers (from 0 to 15)
MAP Multiplexing Scheme	Mission Specific
Truncated Transfer Frame Length	Integer
Inclusion of OCF Allowed (Variable Length Frames)	True ('1'), False ('0')
Inclusion of OCF Required (Fixed Length Frames)	True ('1'), False ('0')
Value for the Repetitions parameter to the Coding Sublayer when transferring Frames carrying service data on the Sequence-Controlled Service	Integer
Value for the Repetitions parameter to the Coding Sublayer when transferring Frames carrying COP control commands	Integer
Maximum delay in milliseconds for a TFDF to be completed, once started, before it must be released	Integer
Maximum delay in milliseconds between releases of Frames of the same VC	Integer

Managed Parameter	Allowed Values
NOTES	
1	The value of the Transfer Frame Length shows either the Frame length (when Transfer Frame type is fixed-length) or the maximum allowed Frame length (when Transfer Frame Type is Variable-Length).
2	VCID value 63 (i.e., the binary 'all ones') is reserved for OID Transfer Frames by 4.1.4.1.5.
3	Transfer Frame Type must equal 'fixed', if either the Physical Channel or Master Channel Transfer Frame Type is fixed.
4	Only when the MC allows variable-length Frames, then the length of a given Transfer Frame is set by the VC Transfer Frame Length above.
5	Table 6-1 lists additional managed parameters required if SDLS is used.

5.5 MANAGED PARAMETERS FOR A MAP CHANNEL

The managed parameters associated with a MAP Channel shall conform to the definitions in table 5-4.

Table 5-4: Managed Parameters for a MAP Channel

Managed Parameter	Allowed Values
MAP ID	One value from a selectable set of integers (from 0 to 15)
Service Data Unit Type	CCSDS Packet, MAPA_SDU, Octet Stream Data
USLP Protocol ID supported	Integer (registered in SANA)

5.6 MANAGED PARAMETERS FOR PACKET TRANSFER

The managed parameters associated with a MAP Channel used for the MAP Packet Service shall conform to the definitions in table 5-5.

Table 5-5: Managed Parameters for Packet Transfer

Managed Parameter	Allowed Values
Valid Packet Version Numbers	Set of Integers
Maximum Packet Length (octets)	Integer
Whether incomplete Packets are required to be delivered to the user at the receiving end	Required, Not required

6 PROTOCOL SPECIFICATION WITH SDLS OPTION

6.1 OVERVIEW

This section specifies the protocol data unit and the procedures of USLP with support for the Space Data Link Security Protocol (reference [14]). If the USLP protocol entity supports SDLS, it has managed parameters for each Virtual Channel to indicate whether SDLS is in use for that channel (see 6.6). Section 4 contains the specification of the protocol without the SDLS option.

6.2 USE OF SDLS PROTOCOL

If SDLS as defined in reference [14] is required over the USLP space data link, then the SDLS protocol shall be used.

NOTE – The SDLS protocol provides a security header and trailer along with associated procedures that may be used with USLP to provide data authentication and data confidentiality at the Data Link Layer.

6.3 USLP TRANSFER FRAME WITH SDLS

6.3.1 OVERVIEW

To support the use of the SDLS security features, a Security Header and a Security Trailer are defined for a USLP Transfer Frame. The use of SDLS can vary between Virtual Channels, so a managed parameter defined in table 6-1 of 6.6 indicates the presence of the Security Header. If the Security Header is present, then SDLS is in use for the Virtual Channel. This subsection specifies the USLP Transfer Frames on a Virtual Channel that is using SDLS.

If a Virtual Channel is not using SDLS, then the Frames are as specified in 4.1.

The Security Header and Security Trailer are placed before and after the Transfer Frame Data Field, and they reduce the length of the Transfer Frame Data Field compared to a Frame without SDLS. Figure 6-1 compares the Frame fields for a Frame without SDLS and a Frame with SDLS. The upper part of figure 6-1 shows the USLP Transfer Frame without the SDLS fields and is the same as figure 4-1.

NOTE – The Frame Error Control Field is controlled by the managed parameters (see section 5).

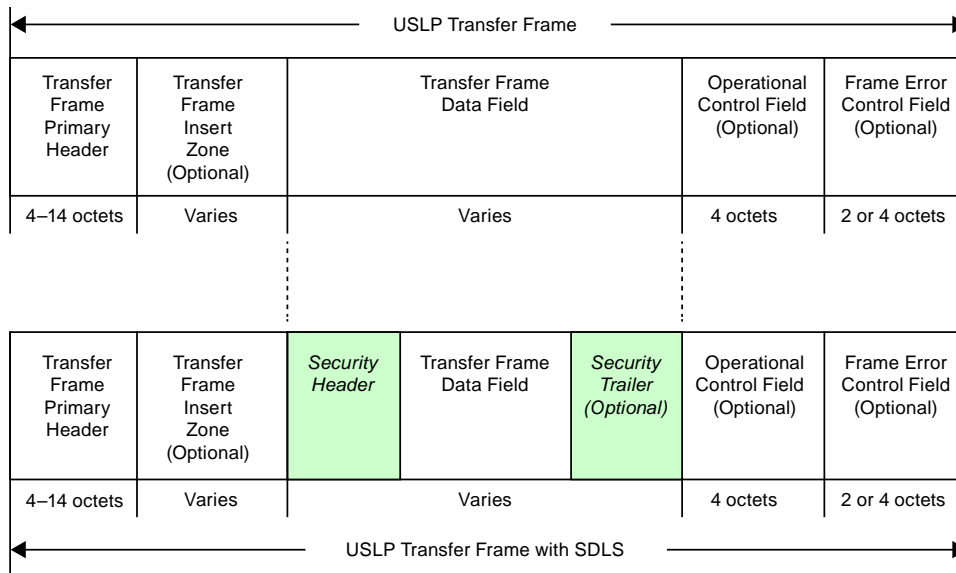


Figure 6-1: Frame without SDLS Compared to Frame with SDLS

6.3.2 TRANSFER FRAME PRIMARY HEADER IN A FRAME WITH SDLS

The Transfer Frame Primary Header for a Frame with SDLS shall conform to the specifications of 4.1.2.

NOTE – The Transfer Frame Primary Header is the same for a Frame without SDLS and a Frame with SDLS.

6.3.3 TRANSFER FRAME INSERT ZONE IN A FRAME WITH SDLS

The Transfer Frame Insert Zone shall conform to the specifications of 4.1.3.

NOTE – The Transfer Frame Insert Zone is the same for a Frame without SDLS and a Frame with SDLS.

6.3.4 SECURITY HEADER

If present, the Security Header shall follow, without gap, the Transfer Frame Insert Zone if a Transfer Frame Insert Zone is present, or the Transfer Frame Primary Header if a Transfer Frame Insert Zone is not present.

NOTES

- 1 The presence of the Security Header is a managed parameter of the Virtual Channel (see 6.6). If the Security Header is not present, the Transfer Frame has the format specified in 4.1.

- 2 The requirements for the length and contents of the Security Header are specified in reference [14].
- 3 The length of the Security Header is an integral number of octets and is a managed parameter of the Virtual Channel.

6.3.5 TRANSFER FRAME DATA FIELD IN A FRAME WITH SDLS

6.3.5.1 The Transfer Frame Data Field of a Frame with SDLS shall conform to the specifications of 4.1.4.1.1 through 4.1.4.1.2 as modified by 6.3.5.2.

6.3.5.2 In a Transfer Frame with SDLS, the Transfer Frame Data Field shall

- a) follow, without gap, the Security Header;

NOTE – Therefore in this case the data unit that is placed into the Transfer Frame Data Field follows, without gap, the Security Header.

- b) contain an integer number of octets equal to the Transfer Frame length, minus
 - 1) the lengths of the Transfer Frame Primary Header and of the Security Header;
 - 2) the lengths of the Transfer Frame Insert Zone, of the Security Trailer, Operational Control Field, and of the Frame Error Control Field, if any of these are present.

6.3.6 SECURITY TRAILER

If present, the Security Trailer shall follow, without gap, the Transfer Frame Data Field.

NOTES

- 1 The Security Trailer is optional in a USLP Transfer Frame with SDLS. The presence of the Security Trailer is a managed parameter of the Virtual Channel (see 6.6).
- 2 The requirements for the length and contents of the Security Trailer are specified in reference [14].
- 3 The length of the Security Trailer is an integral number of octets and is a managed parameter of the Virtual Channel.

6.3.7 OPERATIONAL CONTROL FIELD IN A FRAME WITH SDLS

6.3.7.1 The Operational Control Field of a Frame with SDLS shall conform to the specifications of 4.1.5.2 through 4.1.5.4 as modified by 6.3.7.2.

6.3.7.2 In a Transfer Frame with SDLS, the Operational Control Field, if present, shall occupy the four octets following, without gap, the Security Trailer if this is present, or the Transfer Frame Data Field if a Security Trailer is not present.

6.3.8 FRAME ERROR CONTROL FIELD IN A FRAME WITH SDLS

6.3.8.1 The Frame Error Control Field of a Frame with SDLS shall conform to the specifications of 4.1.6.2, B1, and B1.2, as modified by 6.3.8.2.

6.3.8.2 In a Transfer Frame with SDLS, the Frame Error Control Field, if present, shall occupy from two to four octets following, without gap,

- the Operational Control Field if this is present;
- the Security Trailer if this is present and the Operational Control Field is not present;
- the Transfer Frame Data Field if the Operational Control Field and the Security Trailer are not present.

6.4 SENDING END PROTOCOL PROCEDURES WITH SDLS

6.4.1 OVERVIEW

When a secure USLP link is required, USLP supports the use of the SDLS protocol. In this case, USLP contains differences in the sending end procedures compared to the procedures described in 4.2. This subsection defines those differences.

The SDLS ApplySecurity Function may interface with USLP at either the Virtual Channel Generation Function (4.2.6) or the Virtual Channel Multiplexing Function (4.2.7). The choice of where to apply security within the USLP Data Link Layer depends upon several factors, such as the number of Security Associations (SAs), their type (one VC or more than one VC per SA), and the corresponding source and termination of the security function(s), key management, and the use of the anti-replay feature.

There can be security configurations in which, for example, one or several SAs covering just one VC each are present. The physical location of the security processing may not be the same for all Virtual Channels, at the sending end or at the receiving end. This case can be supported by placing the SDLS interface in the Virtual Channel Generation Function where the greatest flexibility in managing the security function occurs.

6.4.2 PACKET PROCESSING FUNCTION WITH SDLS

6.4.2.1 The Packet Processing Function of a USLP protocol entity that supports SDLS shall conform to the specifications of 4.2.2 and 6.4.2.2.

6.4.2.2 When handling Packets on a Virtual Channel that uses SDLS, the Packet Processing Function shall apply the Transfer Frame Data Field specification in 6.3.5 to determine the length of the TFDZ that it generates.

NOTE – The Packet Processing Function generates a TFDZ to fit exactly within the Transfer Frame Data Field (see 4.1.4).

6.4.3 OCTET STREAM PROCESSING FUNCTION WITH SDLS

6.4.3.1 The Octet Stream Processing Function of a USLP protocol entity that supports SDLS shall conform to the specifications of 4.2.4 and 6.4.3.2.

6.4.3.2 When handling Octet Stream Data on a Virtual Channel that uses SDLS, the Octet Stream Processing Function shall apply the Transfer Frame Data Field specification in 6.3.5 to determine the length of the TFDZ that it generates.

NOTE – The Octet Stream Processing Function generates TFDZs to fit exactly within the Transfer Frame Data Field (see 4.1.4).

6.4.4 VIRTUAL CHANNEL GENERATION FUNCTION WITH SDLS

6.4.4.1 When assembling a Transfer Frame, the Virtual Channel Generation Function shall conform to the specifications of 4.2.6, 6.3, and 6.4.4.2 through 6.4.4.3.

6.4.4.2 The Security Header, and the Security Trailer if it is present for the Virtual Channel, shall be kept empty by the Virtual Channel Generation Function.

NOTES

1 The SDLS ApplySecurity Function specified in reference [14] provides the contents of these security fields as necessary and may modify the contents of the Transfer Frame Data Field by encrypting the data.

2 The lengths of the Security Header and Security Trailer are managed parameters of the Virtual Channel (see 6.6).

6.4.4.3 If the Virtual Channel Generation Function contains the interface to the SDLS protocol,

- a) it shall call the SDLS ApplySecurity function for the Transfer Frames that it assembles for Virtual Channels that use SDLS;
- b) the order of processing between the functions of the USLP and SDLS protocols shall occur as follows in the Virtual Channel Generation Function:
 - 1) the Frame assembly processing by the Virtual Channel Generation Function;

- 2) the call by the Virtual Channel Generation Function to the SDLS ApplySecurity Function.

NOTE – The way in which Transfer Frame data is passed between the Virtual Channel Generation Function and the SDLS ApplySecurity Function is implementation-dependent.

6.4.5 VIRTUAL CHANNEL MULTIPLEXING FUNCTION WITH SDLS

The Virtual Channel Multiplexing Function of a USLP protocol entity that supports SDLS shall conform to the specifications of 4.2.7.

NOTE – There is no interface between the SDLS ApplySecurity function with the USLP ‘Virtual Channel Multiplexing’ function. Multiple data streams may be secured under a single Security Association by defining multiple secure MAP channels within a single VC.

6.4.6 MASTER CHANNEL MULTIPLEXING FUNCTION WITH SDLS

The Master Channel Multiplexing Function of a USLP protocol entity that supports SDLS shall conform to the specifications of 4.2.8.

6.4.7 ALL FRAMES GENERATION FUNCTION WITH SDLS

The All Frames Generation Function of a USLP protocol entity that supports SDLS shall conform to the specifications of 4.2.9.

NOTE – There is no interface between the SDLS ApplySecurity function and the USLP ‘All Frames Generation’ function in order to guarantee that the Frame Error Control field is computed after the SDLS function has processed the Frame.

6.5 RECEIVING END PROTOCOL PROCEDURES WITH SDLS

6.5.1 OVERVIEW

When the USLP Transfer Frame Protocol supports the use of the SDLS protocol, there are differences in the receiving end procedures compared to the procedures described in 4.3. This subsection defines those differences.

The position of the SDLS interface is generally selected to reflect the position of the corresponding interface at the sending end. These choices include the Virtual Channel Demultiplexing Function or the Virtual Channel Reception Function, corresponding to the options discussed in 6.4.1.

6.5.2 ERROR REPORTING

6.5.2.1 Discussion

Depending on the security features in use, the SDLS ProcessSecurity function specified in reference [14] can verify the authenticity of the Frame and it can decrypt the contents of the Transfer Frame Data Field. If the SDLS ProcessSecurity Function detects any errors, these are reported to either the Virtual Channel Demultiplexing Function or the Virtual Channel Reception Function. The way that Transfer Frame data is passed between either of these Functions and the SDLS ProcessSecurity Function is implementation-dependent.

6.5.2.2 Requirements

6.5.2.2.1 If the SDLS ProcessSecurity Function does not report an error, the Virtual Channel Reception Function shall extract the contents of the Transfer Frame Data Field from the Frame and deliver it to its user (or Function).

6.5.2.2.2 If the SDLS ProcessSecurity Function reports an error, either the Virtual Channel Demultiplexing Function or the Virtual Channel Reception Function shall discard the Frame (depending on the interface point).

NOTE – In this case, the optional Verification Status Code parameter can be used to inform the user of the relevant service (see 3.3.2.9, 3.4.2.7, and 3.5.2.6).

6.5.3 PACKET EXTRACTION FUNCTION WITH SDLS

6.5.3.1 The Packet Extraction Function of a USLP protocol entity that supports SDLS shall conform to the specifications of 4.3.2 and 6.5.3.2.

6.5.3.2 When handling Packets on a Virtual Channel that uses SDLS, the Packet Extraction Function shall apply the Transfer Frame Data Field specification in 6.3.5 to determine the expected length of the TFDZs that it receives.

NOTE – The Packet Extraction Function receives TFDZs that fit exactly within the Transfer Frame Data Field (see 4.1.4).

6.5.4 OCTET STREAM EXTRACTION FUNCTION WITH SDLS

6.5.4.1 The Octet Stream Extraction Function of a USLP protocol entity that supports SDLS shall conform to the specifications of 4.2.4 and 6.5.4.2.

6.5.4.2 When handling Octet Stream Data on a Virtual Channel that uses SDLS, the Octet Stream Extraction Function shall apply the Transfer Frame Data Field specification in 6.3.5 to determine the length of the TFDZs that it receives.

NOTE – The Octet Stream Extraction Function receives TFDZs that fit exactly within the Transfer Frame Data Field (see 4.1.4).

6.5.5 VIRTUAL CHANNEL RECEPTION FUNCTION WITH SDLS

6.5.5.1 The Virtual Channel Reception Function of a USLP protocol entity that supports SDLS shall conform to the specifications of 4.3.6 and 6.5.5.2 through 6.5.5.3.

6.5.5.2 If the Virtual Channel Reception Function contains the interface to the SDLS protocol, it shall call the SDLS ProcessSecurity function for the Transfer Frames that it handles for Virtual Channels that use SDLS.

6.5.5.3 When handling a Transfer Frame on a Virtual Channel that uses SDLS, the Virtual Channel Reception Function shall apply the Transfer Frame specification in 6.3 to determine the lengths and positions of the fields in the Transfer Frame.

6.5.6 VIRTUAL CHANNEL DEMULTIPLEXING FUNCTION WITH SDLS

6.5.6.1 The Virtual Channel Demultiplexing Function of a USLP protocol entity that supports SDLS shall conform to the specifications of 4.3.7 and 6.5.6.2.

6.5.6.2 If the Virtual Channel Demultiplexing Function contains the interface to the SDLS protocol, it shall call the SDLS ProcessSecurity function for Transfer Frames on Virtual Channels that use SDLS, before the demultiplexing is applied.

6.5.7 MASTER CHANNEL DEMULTIPLEXING FUNCTION WITH SDLS

The Master Channel Demultiplexing Function of a USLP protocol entity that supports SDLS shall conform to the specifications of 4.3.8.

6.5.8 ALL FRAMES RECEPTION FUNCTION WITH SDLS

The All Frames Reception Function of a USLP protocol entity that supports SDLS shall conform to the specifications of 4.3.9.

6.6 MANAGED PARAMETERS WITH SDLS

6.6.1 OVERVIEW

Managed parameters for the SDLS protocol are specified in reference [14].

6.6.2 ADDITIONAL MANAGED PARAMETERS FOR A VIRTUAL CHANNEL

The managed parameters associated with a Virtual Channel for USLP that supports the SDLS protocol shall conform to the definitions in table 5-3 and the additional definitions in table 6-1.

Table 6-1: Additional Managed Parameters for a Virtual Channel when the Unified Space Data Link Protocol Supports SDLS

Managed Parameter	Allowed Values
Presence of Space Data Link Security Header	Present ('1') / Absent ('0')
Presence of Space Data Link Security Trailer	Present ('1') / Absent ('0')
Length of Space Data Link Security Header (octets)	Integer
Length of Space Data Link Security Trailer (octets)	Integer
NOTES	
1	If the Security Header is present then SDLS is in use for the Virtual Channel.
2	The valid lengths for the Security Header and Security Trailer are specified in reference [14].

ANNEX A

PROTOCOL IMPLEMENTATION CONFORMANCE STATEMENT (PICS) PROFORMA

(NORMATIVE)

A1 INTRODUCTION

A1.1 OVERVIEW

This annex provides the Protocol Implementation Conformance Statement (PICS) Requirements List (RL) for an implementation of the *Unified Space Data Link Protocol* (CCSDS 732.1-R-3). The PICS for an implementation is generated by completing the RL in accordance with the instructions below. An implementation claiming conformance must satisfy the mandatory requirements referenced in the RL.

The RL support column in this annex is blank. An implementation's completed RL is called the PICS. The PICS states which capabilities and options have been implemented. The following can use the PICS:

- the implementer, as a checklist to reduce the risk of failure to conform to the standard through oversight;
- a supplier or potential acquirer of the implementation, as a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the standard PICS proforma;
- a user or potential user of the implementation, as a basis for initially checking the possibility of interworking with another implementation (it should be noted that, while interworking can never be guaranteed, failure to interwork can often be predicted from incompatible PICSes);
- a tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

A1.2 ABBREVIATIONS AND CONVENTIONS

The RL consists of information in tabular form. The status of features is indicated using the abbreviations and conventions described below.

Item Column

The item column contains sequential numbers for items in the table.

NOTE – The item-number prefix 'USLP' = 'Data Link Layer'.

Feature Column

The feature column contains a brief descriptive name for a feature. It implicitly means ‘Is this feature supported by the implementation?’

Status Column

The status column uses the following notations:

M	mandatory.
O	optional.
C#	Conditional; condition stated below table.
O.<n>	optional, but support of at least one of the group of options labeled by the same numeral <n> is required.
N/A	Not applicable.

Support Column Symbols

The support column is to be used by the implementer to state whether a feature is supported by entering Y, N, or N/A, indicating:

Y	Yes, supported by the implementation.
N	No, not supported by the implementation.
N/A	Not applicable.

The support column should also be used, when appropriate, to enter values supported for a given capability.

A1.3 INSTRUCTIONS FOR COMPLETING THE RL

An implementer shows the extent of compliance to the Recommended Standard by completing the RL; that is, the state of compliance with all mandatory requirements and the options supported are shown. The resulting completed RL is called a PICS. The implementer shall complete the RL by entering appropriate responses in the support or values supported column, using the notation described in A1.2. If a conditional requirement is inapplicable, N/A should be used. If a mandatory requirement is not satisfied, exception information must be supplied by entering a reference X_i , where i is a unique identifier, to an accompanying rationale for the noncompliance.

A2 PICS PROFORMA FOR UNIFIED SPACE DATA LINK PROTOCOL (CCSDS 732.1-R-3)

A2.1 GENERAL INFORMATION

A2.1.1 Identification of PICS

Date of Statement (DD/MM/YYYY)	
PICS serial number	
System Conformance statement cross-reference	

A2.1.2 Identification of Implementation Under Test (IUT)

Implementation name	
Implementation version	
Special Configuration	
Other Information	

A2.1.3 Identification of Supplier

Supplier	
Contact Point for Queries	
Implementation Name(s) and Versions	
Other information necessary for full identification, e.g., name(s) and version(s) for machines and/or operating systems;	
System Name(s)	

A2.1.4 Identification of Specification

CCSDS 732.1-R-3	
Have any exceptions been required?	Yes [] No []
NOTE – A YES answer means that the implementation does not conform to the Recommended Standard. Non-supported mandatory capabilities are to be identified in the PICS, with an explanation of why the implementation is non-conforming.	

A2.2 REQUIREMENTS LIST

Table A-1: USLP Service Data Units

Item	Description	Reference	Status	Support
USLP-1	Packet SDU	3.2.2	M	
USLP-2	MAPA SDU	3.2.3	M	
USLP-3	Octet Stream SDU	3.2.4	M	
USLP-4	OCF_SDU	3.2.5	M	
USLP-5	USLP Transfer Frame	3.2.6	M	
USLP-6	Insert Data SDU	3.2.7	M	

Table A-2: Service Parameters

Item	Description	Reference	Status	Values Allowed	Support
MAP Packet Service Parameters					
USLP-7	Packet	3.3.2.2	M		
USLP-8	GMAP ID	3.3.2.3	M		
USLP-9	Packet Version Number	3.3.2.4	M		
USLP-10	SDU ID	3.3.2.5	M		
USLP-11	QoS	3.3.2.6	C1		
USLP-12	Notification Type	3.3.2.7	O	(see ref. [9])	
USLP-13	Packet Quality Indicator	3.3.2.8	O		
USLP-14	Verification Status Code	3.3.2.9	C2	(see ref. [14])	
MAPA Packet Service Parameters					
USLP-15	MAPA_SDU	3.4.2.2	M		
USLP-16	GMAP ID	3.4.2.3	M		
USLP-17	SDU ID	3.4.2.4	M		
USLP-18	QoS	3.4.2.5	C1		
USLP-19	MAPA_SDU Loss Flag	3.4.2.6	O		
USLP-20	Verification Status Code	3.4.2.7	C2	(see ref. [14])	
MAP Octet Stream Service Parameters					
USLP-21	Octet Stream Data	3.5.2.2	M		
USLP-22	GMAP ID	3.5.2.3	M		
USLP-23	QoS	3.5.2.4	C1		
USLP-24	Octet Stream Data Loss Flag	3.5.2.5	O		
USLP-25	Verification Status Code	3.5.2.6	C2	(see ref. [14])	
MC_OCF Service Parameters					
USLP-26	OCF_SDU	3.6.2.2	M		
USLP-27	GVCID	3.6.2.3	M		
USLP-28	OCF_SDU Data Loss Flag	3.6.2.4	O		
VCF Service Parameters					
USLP-29	Frame	3.2.7.2	M		
USLP-30	GVCID	3.2.7.3	M		
USLP-31	Frame Loss Flag	3.2.7.4	O		
MCF Service Parameters					
USLP-32	Frame	3.8.2.2	M		
USLP-33	MCID	3.8.2.3	M		
USLP-34	Frame Loss Flag	3.8.2.4	O		

Item	Description	Reference	Status	Values Allowed	Support
Insert Service Parameters					
USLP-35	IN_SDU	3.9.2.2	M		
USLP-36	Physical Channel Name	3.9.2.3	M		
USLP-37	IN_SDU Loss Flag	3.9.2.4	O		
COPs Management Service Parameters					
USLP-38	GVCID	3.10.2.2.1	O.1		
USLP-39	Port ID	3.10.2.2.2	O.1	(see ref. [10] for COP-P)	
USLP-40	Directive ID	3.10.2.3	M		
USLP-41	Directive Type	3.10.2.4	M	(see refs. [9] for COP-1 and [10] for COP-P)	
USLP-42	Directive Qualifier	3.10.2.5	M	(see refs. [9] for COP-1 and [10] for COP-P)	
USLP-43	Notification Type	3.10.2.6	M	(see refs. [9] for COP-1 and [10] for COP-P)	
USLP-44	Notification Qualifier	3.10.2.7	M	(see refs. [9] for COP-1 and [10] for COP-P)	

C1: M if not separate ports else N/A.

C2: O if SDLS Option else N/A.

O.1: It is mandatory to support one of these options.

Table A-3: Service Primitives

Item	Description	Reference	Status	Support
MAP Packet Service Primitives				
USLP-45	MAPP.request	3.3.3.2	M	
USLP-46	MAPP_Notify.indication	3.3.3.3	M	
USLP-47	MAPP.indication	3.3.3.4	M	
MAPA Service Primitives				
USLP-48	MAPA.request	3.4.3.2	M	
USLP-49	MAPA_Notify.indication	3.4.3.3	M	
USLP-50	MAPA.indication	3.4.3.4	M	
MAP Octet Stream Service Primitives				
USLP-51	OCTET_STREAM.request	3.5.3.2	M	
USLP-52	OCTET_STREAM.indication	3.5.3.3	M	
MC_OCF Service Primitives				
USLP-53	MC_OCF.request	3.6.3.2	M	
USLP-54	MC_OCF.indication	3.6.3.3	M	
VCF Service Primitives				
USLP-55	VCF.request	3.7.3.2	M	
USLP-56	VCF.indication	3.7.3.3	M	
MCF Service Primitives				
USLP-57	MCF.request	3.8.3.2	M	
USLP-58	MCF.indication	3.8.3.3	M	
Insert Service Primitives				
USLP-59	INSERT.request	3.9.3.2	M	
USLP-60	INSERT.indication	3.9.3.3	M	
COPs Management Service Primitives				
USLP-61	Directive.request	3.10.3.2	M	
USLP-62	Directive_Notify.indication	3.10.3.3	M	
USLP-63	Async_Notify.indication	3.10.3.4	M	

Table A-4: USLP Protocol Data Unit

Item	Description	Reference	Status	Support
USLP-64	USLP Transfer Frame	4.1.1	M	
USLP-65	Transfer Frame Primary Header	4.1.2	M	
USLP-66	Transfer Frame Insert Zone	4.1.3	M	
USLP-67	Transfer Frame Data Field	4.1.4	M	
USLP-68	Operational Control Field	4.1.5	M	
USLP-69	Frame Error Control Field	4.1.6	M	

Table A-5: Protocol Procedures

Item	Description	Reference	Status	Support
USLP-70	MAP Packet Processing Function	4.2.2	M	
USLP-71	MAPA_SDU Generation Function	4.2.3	M	
USLP-72	MAP Octet Stream Processing Function	4.2.4	M	
USLP-73	MAP Multiplexing Function	4.2.5	M	
USLP-74	Virtual Channel Generation Function	4.2.6	M	
USLP-75	Virtual Channel Multiplexing Function	4.2.7	M	
USLP-76	Master Channel Multiplexing Function	4.2.8	M	
USLP-77	All Frames Generation Function	4.2.9	M	
USLP-78	MAP Packet Extraction Function	4.3.2	M	
USLP-79	MAPA_SDU Extraction Function	4.3.3	M	
USLP-80	MAP Octet Stream Extraction Function	4.3.4	M	
USLP-81	MAP Demultiplexing Function	4.3.5	M	
USLP-82	Virtual Channel Reception Function	4.3.6	M	
USLP-83	Virtual Channel Demultiplexing Function	4.3.7	M	
USLP-84	Master Channel Demultiplexing Function	4.3.8	M	
USLP-85	All Frames Reception Function	4.3.9	M	

Table A-6: Management Parameters

Item	Description	Reference	Status	Values Allowed	Support
Managed Parameters for a Physical Channel					
USLP-86	Physical Channel Name	Table 5-1	M	Character String	
USLP-87	Transfer Frame Type	Table 5-1	M	Fixed Length or Variable Length	
USLP-88	Transfer Frame Length	Table 5-1	M	Integer (0–65536 octets)	
USLP-89	Transfer Frame Version Number	Table 5-1	M	'1100' binary	
USLP-90	MC Multiplexing Scheme	Table 5-1	M		
USLP-91	Presence of Insert Zone	Table 5-1	M	Present ('1'), Absent ('0')	
USLP-92	Insert Zone Length (octets)	Table 5-1	M	Integer, 1–65514 octets	
USLP-93	Presence of Frame Error Control	Table 5-1	M	Present ('1'), Absent ('0')	
USLP-94	Frame Error Control Length (octets)	Table 5-1	M	2 or 4	
USLP-95	Generate OID Frame	Table 5-1	M	True ('1'), False ('0')	
USLP-96	Maximum Number of Transfer Frames Given to the Coding and Synchronization Sublayer as a single data unit	Table 5-1	M	1	
USLP-97	Maximum Value of the Repetitions Parameter to the Coding and Synchronization Sublayer	Table 5-1	M	Integer	
Managed Parameters for a Master Channel					
USLP-98	Transfer Frame Type	Table 5-2	M	Fixed Length or Variable Length	
USLP-99	Spacecraft ID	Table 5-2	M	16-bit Integer	
USLP-100	VCIDs	Table 5-2	M	0 to 62	
USLP-101	VC Multiplexing Scheme	Table 5-2	M		
Managed Parameters for a Virtual Channel					
USLP-102	Transfer Frame Type	Table 5-3	M	Fixed Length or Variable Length	
USLP-103	VCID	Table 5-3	M	0 to 62	
USLP-104	VC Count Length for Sequence Control	Table 5-3	M	Integer (max. 56 bit)	
USLP-105	VC Count Length for Expedited Integer	Table 5-3	M	Integer (max. 56 bit)	
USLP-106	COP in Effect	Table 5-3	M	COP-1, COP-P, None	
USLP-107	CLCW Version Number	Table 5-3	M	1	

Item	Description	Reference	Status	Values Allowed	Support
USLP-108	CLCW Reporting Rate	Table 5-3	M		
USLP-109	MAP IDs	Table 5-3	M	0–15	
USLP-110	MAP Multiplexing Scheme	Table 5-3	O		
USLP-111	Truncated Transfer Frame Length	Table 5-3	M	Integer	
USLP-112	Inclusion of OCF Allowed (Variable Length Frames)	Table 5-3	M	True ('1'), False ('0')	
USLP-113	Inclusion of OCF Required (Fixed Length Frames)	Table 5-3	M	True ('1'), False ('0')	
USLP-114	Value for the Repetitions parameter to the Coding Sublayer when transferring Frames carrying service data on the Sequence-Controlled Service	Table 5-3	M	Integer (see ref. [6])	
USLP-115	Value for the Repetitions parameter to the Coding Sublayer when transferring Frames carrying COP control commands	Table 5-3	M	Integer (see ref. [6])	
USLP-116	Maximum delay in milliseconds for a TFDF to be completed, once started, before it must be released	Table 5-3	M	Integer	
USLP-117	Maximum delay in milliseconds between releases of Frames of the same VC	Table 5-3	M	Integer	
Managed Parameters for a MAP Channel					
USLP-118	MAP ID	Table 5-4	M	0–15	
USLP-119	Service Data Unit Type	Table 5-4	M	CCSDS Packet, MAPA_SDU, Stream Data	
USLP-120	USLP Protocol ID supported	Table 5-4	M	Integer (see SANA Registry)	
Managed Parameters for a Packet Transfer					
USLP-121	Valid Packet Version Numbers	Table 5-5	M	Set of Integers	
USLP-122	Maximum Packet Length (octets)	Table 5-5	M	Integer	
USLP-123	Whether incomplete Packets are required to be delivered to the user at the receiving end	Table 5-5	M	Required, not required	

Table A-7: Protocol Specification with SDLS Option

Item	Description	Reference	Status	Support
USLP-124	SDLS Protocol	Ref. [14]	O	
USLP-125	Security Header	6.3.4	C3	
USLP-126	Security Trailer	6.3.6	C4	
USLP-127	Transfer Frame Data Field in a Frame with SDLS	6.3.5.2	C3	
USLP-128	Operational Control Field in a Frame with SDLS	6.3.7.2	C3	
USLP-129	Frame Error Control Field in a Frame with SDLS	6.3.8.2	C3	
USLP-130	Packet Processing Function with SDLS	6.4.2.2	C3	
USLP-131	Octet Stream Processing Function with SDLS	6.4.3.2	C3	
USLP-132	Virtual Channel generation Function with SDLS	6.4.4.2, 6.4.4.3	C3	
USLP-133	Error reporting	6.5.2.2	C4	
USLP-134	Packet Extraction Function with SDLS	6.5.3.2	C3	
USLP-135	Octet Stream Extraction Function with SDLS	6.5.4.2	C3	
USLP-136	Virtual Channel Reception Function with SDLS	6.5.5.2, 6.5.5.3	C3	
USLP-137	Virtual Channel Demultiplexing Function with SDLS	6.5.6.2	C3	

C3: M if SDLS Option else N/A.

C4: O if SDLS Option else N/A.

Table A-8: Additional Managed Parameters with SDLS Option

Item	Description	Reference	Status	Values Allowed	Support
USLP-138	Presence of Space Data Link Security Header	Table 6-1	C5	Present ('1') / Absent ('0')	
USLP-139	Presence of Space Data Link Security Trailer	Table 6-1	C5	Present ('1') / Absent ('0')	
USLP-140	Length of Space Data Link Security Header (octets)	Table 6-1	C5	Integer (see ref. [14])	
USLP-141	Length of Space Data Link Security Trailer (octets)	Table 6-1	C5	Integer (see ref. [14])	

C5: M if SDLS Option else N/A.

Table A-9: Frame Error Control Field Coding Procedures

USLP-142	CRC-16 FECF Encoding Procedure	B1.1	M	
USLP-143	CRC-16 FECF Decoding Procedure	B1.2	M	
USLP-144	CRC-32 Encoding Procedure	B2.1	M	
USLP-145	CRC-32 FECF Decoding Procedure	B2.2	M	

Table A-10: Relationship of Version-3 and Version-4 Transfer Frames

USLP-146	V3-V4 equivalencies	Annex C	C6	
----------	---------------------	---------	----	--

C6: M if gateway else N/A.

ANNEX B

FRAME ERROR CONTROL FIELD CODING PROCEDURES

(NORMATIVE)

B1 CRC-16 FRAME ERROR CONTROL FIELD CODING PROCEDURES

B1.1 CRC-16 FECF ENCODING PROCEDURE

B1.1.1 The Frame Error Control Field shall be computed by applying Cyclic Redundancy Check (CRC) techniques.

B1.1.2 The Encoding Procedure shall accept an $(n-16)$ -bit Transfer Frame, excluding the Frame Error Control Field, and generate a systematic binary $(n, n-16)$ block code by appending a 16-bit Frame Error Control Field as the final 16 bits of the codeword, where n is the length of the Transfer Frame.

NOTE – The Bit Numbering Convention specified in 1.6.3 is applicable below.

B1.1.3 The equation for the contents of the Frame Error Control Field is:

$$\begin{aligned} \text{FECF} &= [(X^{16} \cdot M(X)) + (X^{(n-16)} \cdot L(X))] \text{ modulo } G(X) \\ &= P_0 \cdot X^{15} + P_1 \cdot X^{14} + P_2 \cdot X^{13} + \dots + P_{14} \cdot X^1 + P_{15} \cdot X^0 \end{aligned}$$

where

all arithmetic is modulo 2;

FECF is the 16-bit Frame Error Control Field with the first bit transferred being the most significant bit P_0 taken as the coefficient of the highest power of X ;

n is the number of bits in the encoded message;

$M(X)$ is the $(n-16)$ -bit information message to be encoded expressed as a polynomial with binary coefficients, with the first bit transferred being the most significant bit M_0 taken as the coefficient of the highest power of X ;

$L(X)$ is the presetting polynomial given by

$$L(X) = \sum_{i=0}^{15} X^i ;$$

$G(X)$ is the generating polynomial given by

$$G(X) = X^{16} + X^{12} + X^5 + 1.$$

NOTES

- 1 The $X^{(n-16)} \cdot L(X)$ term has the effect of presetting the shift register to all '1' state prior to encoding.
- 2 A possible FECF generator implementation is shown in figure B-1. For each Frame, the shift register cells are initialized to '1'. The ganged switch is in position 1 while the information bits are being transferred and in position 2 for the sixteen FECF bits.

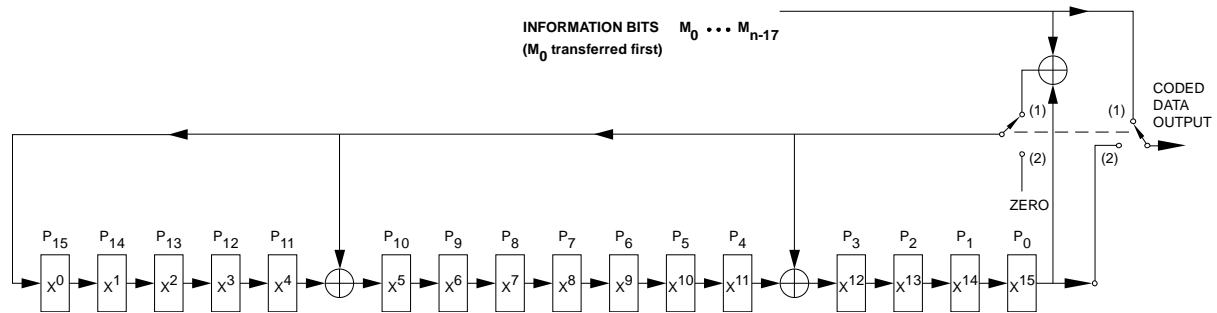


Figure B-1: Logic Diagram of the CRC-16 Encoder

B1.2 CRC-16 FECF DECODING PROCEDURE

B1.2.1 The error detection syndrome, $S(X)$, is given by

$$S(X) = [(X^{16} \cdot C^*(X)) + (X^n \cdot L(X))] \text{ modulo } G(X)$$

where

- $C^*(X)$ is the received block, including the Frame Error Control Field, in polynomial form, with the first bit transferred being the most significant bit C_0^* taken as the coefficient of the highest power of X ; and
- $S(X)$ is the syndrome polynomial, which will be zero if no error is detected and non-zero if an error is detected, with the most significant bit S_0 taken as the coefficient of the highest power of X .

B1.2.2 The received block $C^*(X)$ must equal the transmitted codeword $C(X)$ plus (modulo two) the n -bit error block $E(X)$, $C^*(X) = C(X) + E(X)$, where both are expressed as polynomials of the same form, i.e., with the most significant bit C_0 or E_0 taken as the binary coefficient of the highest power of X .

NOTE – A possible syndrome polynomial generator implementation is shown in figure B-2. For each Frame, the shift register cells are initialized to ‘1’. The Frame includes n bits, i.e., $(n-16)$ information message bits plus the 16 bits of the FECF. All the n bits of the Frame are clocked into the input, and then the storage stages are examined. For an error-free block, the contents of the shift register cells will be ‘zero’. A non-zero content indicates an erroneous block.

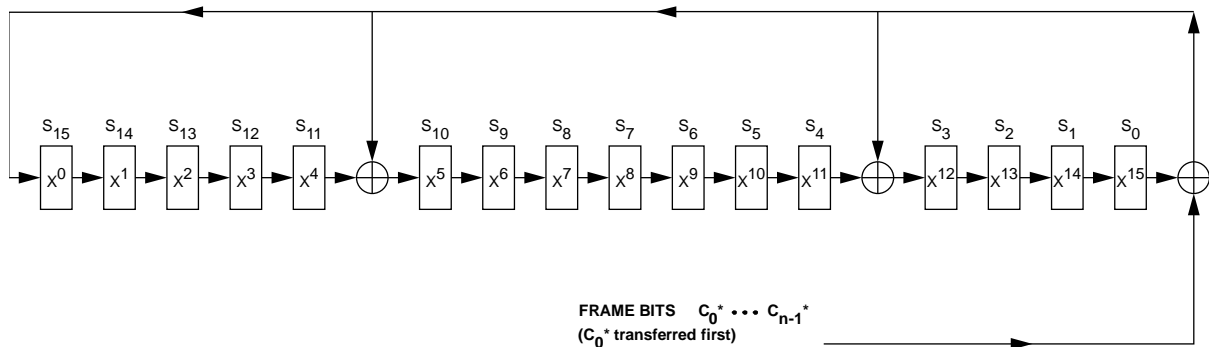


Figure B-2: Logic Diagram of the CRC-16 Decoder

B2 CRC-32 FRAME ERROR CONTROL FIELD CODING PROCEDURES

B2.1 CRC-32 ENCODING PROCEDURE

B2.1.1 For the encoding procedure, the $(n-32)$ -bit Transfer Frame shall be the information message.

B2.1.2 The encoding procedure shall accept an $(n-32)$ -bit Transfer Frame and generate a systematic binary $(n, n-32)$ block code by appending a 32-bit Cyclic Redundancy Check (CRC-32) as the final 32 bits of the FECF.

NOTES

- 1 The Bit Numbering Convention as specified in 1.6.3 is applicable below.
- 2 The Attached Sync Marker (ASM) is NOT used for computing the CRC-32.

B2.1.3 If $M(X)$ is the $(n-32)$ -bit information message to be encoded, expressed as a polynomial with binary coefficients with the first bit transferred being the most significant bit M_0 taken as the coefficient of the highest power of X , then the equation for the 32-bit Cyclic Redundancy Check, expressed as a polynomial $R(X)$ with binary coefficients, shall be:

$$R(X) = [X^{32} \cdot M(X)] \text{ modulo } G(X)$$

where $G(X)$ is the generating polynomial given by:

$$G(X) = X^{32} + X^{23} + X^{21} + X^{11} + X^2 + 1$$

and where the first transferred bit of the Cyclic Redundancy Check is the most significant bit R_0 taken as the coefficient of the highest power of X .

B2.1.4 The n -bit CRC-32-encoded block, expressed as a polynomial $C(X)$ with binary coefficients, shall be:

$$C(X) = X^{32} \cdot M(X) + R(X)$$

B2.1.5 The $(n-32)$ bits of the message shall be input in the order M_0, \dots, M_{n-33} , and the n bits of the codeword are output in the order $C_0, \dots, C_{n-1} = M_0, \dots, M_{n-33}, R_0, \dots, R_{31}$.

NOTE – A possible implementation of an encoder is described in figure B-3. For each Frame, the shift register is preset to the ‘all zero’ state prior to encoding. This initialization differs from that performed for the 16-bit CRC described in other CCSDS books, for which the cells are initialized to all ‘ones’. The ganged switch is in position 1 while the information bits are being transferred and in position 2 for the 32 Cyclic Redundancy Check bits.

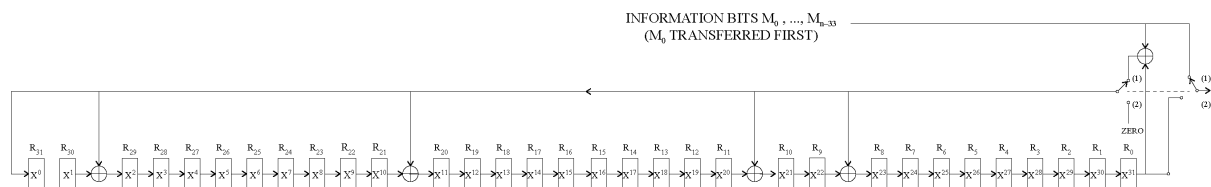


Figure B-3: A Possible Implementation of the CRC-32 Encoder

B2.2 CRC-32 FECF DECODING PROCEDURE

B2.2.1 The decoding procedure shall accept an n -bit received codeword, including the 32-bit Cyclic Redundancy Check, and generate a 32-bit syndrome.

B2.2.2 An error shall be detected if and only if at least one of the syndrome bits is non-‘zero’.

B2.2.3 The received block $C^*(X)$ shall equal the transmitted codeword $C(X)$ plus (modulo two) the n -bit error block $E(X)$, $C^*(X) = C(X) + E(X)$, where both are expressed as polynomials of the same form, i.e., with the most significant bit C_0 or E_0 taken as the binary coefficient of the highest power of X .

B2.2.4 With $C^*(X)$ being the n -bit received codeword with the first transferred bit being the most significant bit C_0^* taken as the coefficient of the highest power of X , the equation for the 32-bit syndrome, expressed as a polynomial $S(X)$ with binary coefficients, shall be:

$$S(X) = [X^{32} \cdot C^*(X)] \text{ modulo } G(X)$$

NOTES

- 1 The syndrome polynomial will be 'zero' if no error is detected and non-'zero' if an error is detected, with the most significant bit S_0 taken as the coefficient of the highest power of X .
- 2 A possible implementation of the syndrome polynomial generator is described in figure B-4. For each Frame, the shift register cells are initialized to 'zero'. This initialization differs from that performed for the 16-bit CRC described in other CCSDS books, for which the cells are initialized to 'all ones'. The codeword includes n bits, i.e., $(n-32)$ information message bits plus the 32 bits of the Cyclic Redundancy Check. All the n bits of the codeword are clocked into the input and then the storage stages are examined. For an error-free block, the contents of the shift register cells will be zero. A non-zero content indicates an erroneous block.

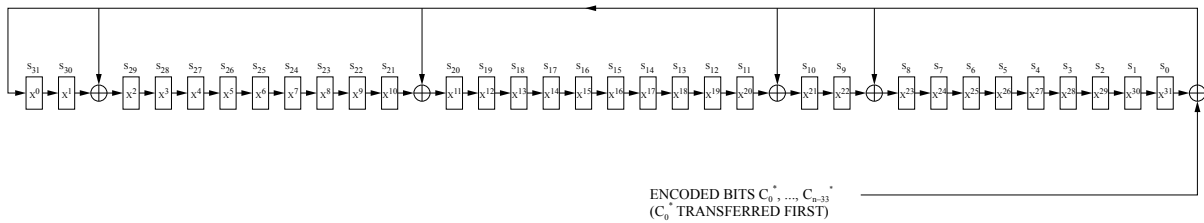


Figure B-4: A Possible Implementation of the CRC-32 Decoder

ANNEX C

**RELATIONSHIP OF PROXIMITY-1 (VERSION-3) TO USLP
(VERSION-4) TRANSFER FRAMES****(NORMATIVE)****C1 TRANSFER FRAME PRIMARY HEADER****C1.1 GENERAL**

Version-3 Transfer Frame header fields shall be mapped to Version-4 Transfer Frame primary header fields as specified in table C-1 below.

Table C-1: Relationship of Transfer Frame Header Fields between Version-3 and Version-4 Frames

Version-3 Frame (Proximity-1)		Version-4 Frame (USLP)	
Field Name	Location in Frame	Field Name	Location in Frame
Transfer Frame Version Number	Bits 0–1	Transfer Frame Version Number	Bits 0–3
Quality of Service Indicator	Bit 2	Bypass Flag	Bit 48
PDU Type ID	Bit 3	Protocol Control Command Flag	Bit 49
Data Field Construction ID (DFC ID)	Bits 4–5	TFDZ Construction Rules in TFDZ Header	(see table C-2)
Spacecraft Identifier	Bits 6–15	Spacecraft Identifier	Bits 4–19
Physical Channel Identifier (PCID)	Bit 16	Virtual Channel Identifier	Use only Bit 21
N/A	N/A	Virtual Channel Identifier	Use Bits 22–26
Port ID	Bits 17–19	MAP ID	Use only Bits 28–30
Source-or-Destination Identifier	Bit 20	Source-or-Destination Identifier	Bit 20
Frame Length	Bits 21–31	Frame Length	Bits 32–47
Frame Sequence Number	Bits 32–39	Virtual Channel Frame Count	Bits 56–63
N/A	N/A	End of Frame Primary Header Flag	Bit 31
N/A	N/A	OCF Flag	Bit 52
N/A	N/A	VC Frame Count Length	Bit 53–55

C1.2 TRANSFER FRAME VERSION NUMBER

NOTE – The Version-3 TFDV has been extended by two bits to allow for future versions.

The USLP Transfer Frame shall use the last remaining value available in the 2-bit Transfer Frame Version Number field, i.e., '11', and append '00' to it to complete the 4-bit Version-4 TFDV ('1100').

C1.3 QUALITY OF SERVICE INDICATOR

The Bypass Flag in bit 48 of the Version-4 Transfer Frame Primary Header shall be equivalent to the Quality of Service Indicator field in bit 2 of the Version-3 Frame.

C1.4 PDU TYPE ID

The Protocol Control Command Flag in bit 49 of the Version-4 Transfer Frame Primary Header shall be equivalent to the PDU Type Indicator in bit 3 of the Version-3 Frame.

C1.5 DATA FIELD CONSTRUCTION ID (DFC ID)

The Transfer Frame Data Zone (TFDZ) Construction Rules in bits 1–3 of the Version-4 TFDZ Header shall replace the functionality of both the Data Field Construction ID in bits 4–5 of the Version-3 Frame as well as the Segment Header Sequence Flags defined in the first two bits of the Proximity-1 Segment Header within the Version-3 Transfer Frame Data Field. (Table C-2 below relates the contents of the Version-3 TFDZ to the Version-4 TFDZ.)

Table C-2: TFDZ Construction Rules

Version-3 DFC ID	Version-3 DFC ID Interpretation	Version-3 Segment Header Sequence Flags	Version-4 TFDZ Construction Rule Equivalent	Version-4 TFDZ Construction Rule Interpretation
'00'	Packets (integer number of unsegmented packets)	N/A	'111'	No Segmentation
'01'	Segment Data (a complete or segmented packet)	'00' '01' '10' '11'	'100' '101' '110' '111'	Starting Segment Continuing Segment Last Segment No Segmentation
'10'	Reserved for future CCSDS definition	N/A	N/A	N/A
'11'	User Defined Data	N/A	'011'	Octet Stream

C1.6 SPACECRAFT ID

The Spacecraft ID in bits 4–19 of the Version-4 Transfer Frame Primary Header shall replace the Spacecraft ID in bits 6–15 of the Version-3 Transfer Frame.

NOTE – The Spacecraft ID field in the Version-4 Frame is specified as a 16 bit field to accommodate the increase in SCID requests for future missions.

C1.7 PHYSICAL CHANNEL ID

The most significant bit of the Virtual Channel ID field (bit 21) of the Version-4 Transfer Frame Primary Header shall be used for the Physical Channel ID in the Version-3 Frame header.

NOTES

- 1 The VCID in the Version-4 Transfer Frame is a 6-bit field, but only the first bit of this field is utilized for the Physical Channel ID.
- 2 The use of a separate VCID for Expedited Frames eliminates the need to have two separate Virtual Channel Frame Counters for the PCID.

C1.8 PORT IDENTIFIER (PORT ID)

The MAP ID in bits 28–30 of the Version-4 Transfer Frame Primary Header shall replace the Port ID in bits 17–19 of the Version-3 Frame.

C1.9 SOURCE/DESTINATION ID

The source or destination ID in bit 20 of the Version-4 Transfer Frame Primary Header shall be equivalent to the Source/Destination ID in bit 20 of the Version-3 Frame.

C1.10 DISCUSSION—FRAME LENGTH

The Frame Length Field in the Version-4 Transfer Frame Primary Header is 16 bits in length and is located in bits 32–47. The Frame Length Field in the Version-3 Frame is 11 bits in length and located in bits 21–31. The Version-4 Frame can support Frame lengths up to 65536 octets.

C1.11 VIRTUAL CHANNEL FRAME COUNT LENGTH

For Proximity-1 operations, the value of the Virtual Channel Frame Count Length Field in the Version-4 Transfer Frame Primary Header shall be set to ‘001’, i.e., one octet.

C1.12 FRAME SEQUENCE NUMBER

The Virtual Channel Frame Count in bits 56–63 of the Version-4 Transfer Frame Primary Header shall replace the Frame sequence counter in bits 32–39 of the Version-3 Frame.

C1.13 TRUNCATED TRANSFER FRAME PRIMARY HEADER

The truncated Frame shall be created and signaled by setting the End of Frame Primary Header Flag (bit 31) to '1'.

NOTES

- 1 The truncated primary header option is provided for short hardware commands. Since the Frame is truncated and has no length field, the Frame length is controlled by a managed parameter and is set for the mission.
- 2 Because the Transfer Frame Primary Header is truncated and the length field is not present, the Frame size is fixed based upon the value contained in the Truncated Transfer Frame Length managed parameter.

C2 DISCUSSION—INSERT ZONE

The Insert Zone is specified for use with fixed-length Frames of constant length that are aligned with fixed-length codeblocks. Thus the use of the Insert Zone in proximity operations is not allowed.

C3 TRANSFER FRAME DATA FIELD**C3.1 OVERVIEW**

The TFDF Header for Proximity operations utilizes a subset of the fields defined in USLP. Because Proximity-1 is defined to use variable-length Frames exclusively, the USLP Pointer Field is not needed in the TFDF Header. Thus the TFDF Header reduces to a header composed of two fields immediately followed by the Transfer Frame Data Zone (TFDZ).

TFDZ Construction Rules	USLP Protocol Identifier
3 bits	5 bits

NOTE – The Pointer Field (see figure 4-5) is not applicable for Proximity operations.

Figure C-1: Transfer Frame Data Field Header for Proximity Operations

C3.2 TFDZ HEADER

C3.2.1 General

For proximity operations, the TFDZ Header shall utilize the first two fields of the TFDZ defined in USLP:

- a) the TFDZ Construction Rules, 3 bits;
- b) UPID, 5 bits.

C3.2.2 TFDZ Construction Rules

The TFDZ Construction Rules that apply to Proximity-1 operations shall be those specified in table C-2 above.

NOTE – The TFDZ Construction Rules for proximity operations utilizes a subset of the eight rules defined in USLP. Three of these rules are defined exclusively for fixed-length Frames and thus are not applicable for Proximity-1 operations. Segmentation for Proximity operations is accomplished by using the TFDZ Construction Rules. Therefore the Segment Header as defined in Proximity-1 as the first octet of the Transfer Frame Data field is not used.

C3.2.3 USLP Protocol ID

The UPID applicable to Proximity-1 operations shall be interpreted as follows:²

- a) The value '00000' in the UPID field signals that Space Packets or Encapsulation packets are contained within the TFDZ.
- b) The value '00010' in the UPID field signals that COP-P directives are contained within the TFDZ.
- c) The value '00011' in the UPID field signals that SDLS directives are contained within the TFDZ.
- d) The value '00100' in the UPID field signals that user-defined stream data are contained within the TFDZ.
- e) The value '00101' in the UPID field signals that mission-specific-information-1 is contained within the TFDZ.
- f) The value '00110' in the UPID field signals that mission-specific-information-2 is contained within the TFDZ.
- g) The value '00111' in the UPID field signals that Proximity-1 Supervisory Protocol Data Units (SPDUs) are contained within the TFDZ.

² When this Recommended Standard is finalized, the USLP Protocol IDs will be registered and maintained within SANA.

- h) The value '01000' in the UPID field signals that either a partial or complete MAPA_SDU is contained within the TFDZ.

NOTE – There is no UPID value for a PLCW, since when present the PLCW resides in the OCF.

C4 DISCUSSION—SECURITY HEADER AND TRAILER

The presence of the Security Header and Security Trailer is controlled by the USLP Virtual Channel managed parameters. It is anticipated that these managed parameters will be fixed for a mission. Because USLP defines the Virtual Channel ID field, and because the most significant bit of the VCID field for Proximity-1 operations is defined as the Physical Channel ID equivalent (see C1.7), there are only 32 VCIDs defined for Proximity Link operations over USLP.

NOTE – The VCID associated with a specific spacecraft can be set for the mission and different spacecraft could be assigned different VCIDs so that they could utilize very different Security Associations as defined in SDLS.

C5 OCF

The OCF may be used to transfer either a PLCW or an SDLS Frame Security Report (FSR).

NOTES

- 1 The OCF Flag in bit 52 of the Version-4 Transfer Frame Primary Header signals the presence or absence of the OCF field in the Frame. The use of the OCF is optional.
- 2 For Proximity-1 operations using the Version-4 Frame, a PLCW or FSR can be transferred either
 - as a Protocol Control Command, i.e., in Proximity-1 terms, as an SPDU sent in a separate P-frame; or
 - as user data within the OCF.
- 3 The PLCW or FSR can be sent in the OCF field of a Transfer Frame that does not contain a TFDZ.

C6 FECF

The 4-octet FECF is mandatory for Proximity-1 operations. (See reference [7].)

NOTE – Unlike the Proximity-1 Frame, the USLP Transfer Frame defines the FECF within the Frame and thus the Frame length includes these four octets.

ANNEX D

SECURITY, SANA, AND PATENT CONSIDERATIONS

(INFORMATIVE)

D1 SECURITY CONSIDERATIONS

Security Considerations at the Data Link Layer are addressed by the *Space Data Link Security Protocol* (reference [14]).

D2 SANA CONSIDERATIONS

D2.1 GENERAL

This section contains new and modified SANA registry requests for USLP.

D2.2 NEW REGISTRY REQUESTS

D2.2.1 USLP Protocol Identifier

UPID = '00000' to '11111'

Status = Managed

Reference: CCSDS 732.1-R-3

D2.3 MODIFIED REGISTRIES

D2.3.1 CCSDS Abbreviations

Abbreviation: USLP – Unified Space Data Link Protocol

Reference: CCSDS 732.1-R-3

D2.3.2 CCSDS Terms

isochronous: Characterized by occurrence at equal intervals of time.

D2.3.3 MAP ID

For TFVN = 4, the MAP ID is constrained to values '0000' – '1111'.

D2.3.4 Spacecraft Identifier

NOTE – Version 4 (VN = 1100); SCID = 16 bits; GSCID = VN.SCID = 20 bits;

Version 4 Range: SCID = [0x0000 – 0xFFFF]; GSCID = [C0000 - CFFFF].

D2.3.5 Transfer Frame Version Number

Version Number = ‘1100’; Transfer Frame = USLP Version 4; Reference = CCSDS 732.1-R-3

D2.3.6 VCID

Add USLP Protocol to the registry for VCID:

VCID = 000000 – 111111

Status: Unmanaged

Reference: CCSDS 732.1-R-3

D3 PATENT CONSIDERATIONS

At the time of publication, CCSDS was not aware of any claimed patent rights applicable to implementing the provisions of this Recommended Standard.

ANNEX E

INFORMATIVE REFERENCES

(INFORMATIVE)

- [E1] *Organization and Processes for the Consultative Committee for Space Data Systems*. Issue 4. CCSDS Record (Yellow Book), CCSDS A02.1-Y-4. Washington, D.C.: CCSDS, April 2014.
- [E2] *Overview of Space Communications Protocols*. Issue 3. Report Concerning Space Data System Standards (Green Book), CCSDS 130.0-G-3. Washington, D.C.: CCSDS, July 2014.
- [E3] *Space Communications Cross Support—Architecture Description Document*. Issue 1. Report Concerning Space Data System Standards (Green Book), CCSDS 901.0-G-1. Washington, D.C.: CCSDS, November 2013.
- [E4] *Space Communications Cross Support—Architecture Requirements Document*. Issue 1. Recommendation for Space Data System Practices (Magenta Book), CCSDS 901.1-M-1. Washington, D.C.: CCSDS, May 2015.
- [E5] *Cross Support Reference Model—Part 1: Space Link Extension Services*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 910.4-B-2. Washington, D.C.: CCSDS, October 2005.
- [E6] *TC Space Data Link Protocol*. Issue 3. Recommendation for Space Data System Standards (Blue Book), CCSDS 232.0-B-3. Washington, D.C.: CCSDS, September 2015.
- [E7] *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.
- [E8] *Digital Video Broadcasting (DVB); Framing Structure, Channel Coding and Modulation for 11/12 GHz Satellite Services*. ETSI EN 300 421 V1.1.2 (1997-08). Sophia-Antipolis: ETSI, 1997.
- [E9] *A 48/56/64 kbit/s Data Circuit-Terminating Equipment Standardized for Use on Digital Point-to-Point Leased Circuits*. ITU-T Recommendation V.38. Geneva: ITU, 1996.
- [E10] *Performance Characteristics for Intermediate Data Rate Digital Carriers Using Convolutional Encoding/Viterbi Encoding*. Rev. 10. IESS 308. Washington, DC: INTELSAT, 2000.

- [E11] *Unified Space Link Protocol*. Report Concerning Space Data System Standards (Green Book), CCSDS 700.1-G. Forthcoming.
- [E12] *Advanced Orbiting Systems, Networks and Data Links: Summary of Concept, Rationale and Performance*. Issue 3. Report Concerning Space Data System Standards (Green Book), CCSDS 700.0-G-3. Washington, D.C.: CCSDS, November 1992.
- [E13] *TC Synchronization and Channel Coding—Summary of Concept and Rationale*. Issue 2. Report Concerning Space Data System Standards (Green Book), CCSDS 230.1-G-2. Washington, D.C.: CCSDS, November 2012.
- [E14] *Proximity-1 Space Link Protocol—Rationale, Architecture, and Scenarios*. Issue 2. Report Concerning Space Data System Standards (Green Book), CCSDS 210.0-G-2. Washington, D.C.: CCSDS, December 2013.
- [E15] *Telecommand Summary of Concept and Rationale*. Issue 6. Report Concerning Space Data System Standards (Green Book), CCSDS 200.0-G-6. Washington, D.C.: CCSDS, January 1987.
- [E16] *TM Synchronization and Channel Coding—Summary of Concept and Rationale*. Issue 2. Report Concerning Space Data System Standards (Green Book), CCSDS 130.1-G-2. Washington, D.C.: CCSDS, November 2012.
- [E17] *Next Generation Uplink*. Issue 1. Report Concerning Space Data System Standards (Green Book), CCSDS 230.2-G-1. Washington, D.C.: CCSDS, July 2014.
- [E18] *Space Data Link Protocols—Summary of Concept and Rationale*. Issue 3. Report Concerning Space Data System Standards (Green Book), CCSDS 130.2-G-3. Washington, D.C.: CCSDS, September 2015.
- [E19] *TM Space Data Link Protocol*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 132.0-B-2. Washington, D.C.: CCSDS, September 2015.
- [E20] *AOS Space Data Link Protocol*. Issue 3. Recommendation for Space Data System Standards (Blue Book), CCSDS 732.0-B-3. Washington, D.C.: CCSDS, September 2015.
- [E21] *Proximity-1 Space Link Protocol—Physical Layer*. Issue 4. Recommendation for Space Data System Standards (Blue Book), CCSDS 211.1-B-4. Washington, D.C.: CCSDS, December 2013.
- [E22] *Time Code Formats*. Issue 4. Recommendation for Space Data System Standards (Blue Book), CCSDS 301.0-B-4. Washington, D.C.: CCSDS, November 2010.

NOTE – Normative references are listed in 1.7.

ANNEX F**ABBREVIATIONS AND ACRONYMS****(INFORMATIVE)**

This annex lists the acronyms used in this Recommended Standard.

<u>Term</u>	<u>Meaning</u>
AOS	Advanced Orbiting Systems
ARQ	automatic repeat queuing
ASM	Attached Sync Marker
CC	channel coding
CCSDS	Consultative Committee for Space Data Systems
CLCW	Communications Link Control Word
CLTU	Communications Link Transmission Unit
COP-1	Communications Operation Procedure 1
COP-P	Communications Operation Procedure Proximity
DVB-S2	Digital Video Broadcasting - Satellite - Second Generation
FARM	Frame Acceptance and Reporting Mechanism
FARM-B	Frame Acceptance and Reporting Mechanism Expedited Service
FARM-P	Frame Acceptance and Reporting Mechanism-Proximity
FECF	Frame Error Control Field
FHP	First Header Pointer
FOP	Frame Operation Procedure
FSR	Frame Security Report
GMAP	Global Multiplexer Access Point
GMAP ID	Global Multiplexer Access Point Identifier
GSCID	Global Spacecraft Identifier
GVCID	Global Virtual Channel Identifier
IESS	Intelsat Earth Station Standards
IN_SDU	Insert Service Data Unit
MAP	Multiplexer Access Point
MAP ID	Multiplexer Access Point Identifier

<u>Term</u>	<u>Meaning</u>
MAPA	Multiplexer Access Point Access
MAPA_SDU	Multiplexer Access Point Access Service Data Unit
MAPP	Multiplexer Access Point Packet
MC	Master Channel
MC_OCF	Master Channel Operational Control Field
MCF	Master Channel Frame
MCID	Master Channel Identifier
MSB	most significant bit
OCF	Operational Control Field
OCF_SDU	Operational Control Field Service Data Unit
OID	Only Idle Data (Transfer Frame)
OSI	Open Systems Interconnection
PCID	Physical Channel ID
PDU	protocol data unit
PICS	protocol implementation conformance statement
PICS	protocol implementation conformance statement
PLCW	Proximity Link Control Word
PVN	Packet Version Number
QoS	quality of service
RF	radio frequency
SA	Security Association
SANA	Space Assigned Numbers Authority
SAP	service access point
SCCC	Serially Concatenated Convolutional Code
SCID	spacecraft identifier
SDLS	Space Data Link Security
SDU	service data unit
SLP	space link protocol
SPDU	Supervisory Protocol Data Unit
Sync	Synchronization
TC	telecommand

<u>Term</u>	<u>Meaning</u>
TFDF	Transfer Frame Data Field
TFDZ	Transfer Frame Data Zone
TFVN	Transfer Frame Version Number
TM	telemetry
UPID	USLP Protocol ID
USLP	Unified Space Data Link Protocol
VC	Virtual Channel
VCF	Virtual Channel Frame
VCID	Virtual Channel Identifier
VN	version number