CCSDS Historical Document

This document's Historical status indicates that it is no longer current. It has either been replaced by a newer issue or withdrawn because it was deemed obsolete. Current CCSDS publications are maintained at the following location:

http://public.ccsds.org/publications/

Consultative Committee for Space Data Systems

RECOMMENDATION FOR SPACE

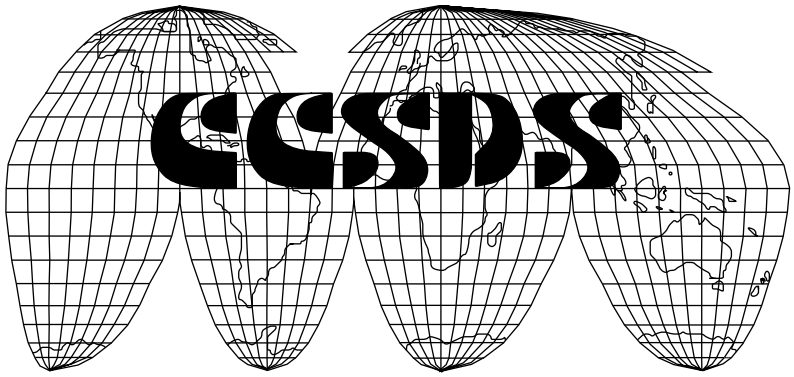
DATA SYSTEM STANDARDS

PACKET TELEMETRY SERVICE SPECIFICATION

CCSDS 103.0-B-2

BLUE BOOK

June 2001



AUTHORITY

Issue:	Blue Book, Issue 2
Date:	June 2001
Location:	Oxfordshire, UK

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 Recommendations is detailed in reference [B1], and the record of Agency participation in the authorization of this document can be obtained from the CCSDS Secretariat at the address below.

This Recommendation is published and maintained by:

CCSDS Secretariat
Program Integration Division (Code MT)
National Aeronautics and Space Administration
Washington, DC 20546, USA

STATEMENT OF INTENT

The Consultative Committee for Space Data Systems (CCSDS) is an organization officially established by the management of member space Agencies. 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 **Recommendations** and are not considered binding on any Agency.

This **Recommendation** is issued by, and represents the consensus of, the CCSDS Plenary body. Agency endorsement of this **Recommendation** is entirely voluntary. Endorsement, however, indicates the following understandings:

- Whenever an Agency establishes a CCSDS-related **standard**, this **standard** will be in accord with the relevant **Recommendation**. Establishing such a **standard** does not preclude other provisions which an Agency may develop.

- Whenever an Agency establishes a CCSDS-related **standard**, the Agency will provide other CCSDS member Agencies with the following information:

  -- The **standard** itself.

  -- The anticipated date of initial operational capability.

  -- The anticipated duration of operational service.

- Specific service arrangements shall be made via memoranda of agreement. Neither this **Recommendation** nor any ensuing **standard** is a substitute for a memorandum of agreement.

No later than five years from its date of issuance, this **Recommendation** 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 **Recommendation** is issued, existing CCSDS-related Agency standards and implementations are not negated or deemed to be non-CCSDS compatible. It is the responsibility of each Agency to determine when such standards or implementations are to be modified. Each Agency is, however, strongly encouraged to direct planning for its new standards and implementations towards the later version of the Recommendation.

FOREWORD

This document is a technical **Recommendation** for use in developing packetized telemetry systems and has been prepared by the **Consultative Committee for Space Data Systems** (CCSDS). The Packet Telemetry Services described herein are intended for spacecraft-to-ground data communication within missions that are cross-supported between Agencies of the CCSDS.

This **Recommendation** establishes a common framework and provides a common basis for the data services of spacecraft telemetry systems. It allows implementing organizations within each Agency to proceed coherently with the development of compatible derived Standards for the flight and ground systems that are within their cognizance. Derived Agency Standards may implement only a subset of the optional features allowed by the **Recommendation** and may incorporate features not addressed by the **Recommendation**.

Through the process of normal evolution, it is expected that expansion, deletion or modification to this document may occur. This **Recommendation** is therefore subject to CCSDS document management and change control procedures which are defined in reference [B1].

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

Member Agencies

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

Observer Agencies

- Austrian Space Agency (ASA)/Austria.
- Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
- Centro Tecnico Aeroespacial (CTA)/Brazil.
- Chinese Academy of Space Technology (CAST)/China.
- Commonwealth Scientific and Industrial Research Organization (CSIRO)/Australia.
- Communications Research Centre (CRC)/Canada.
- Communications Research Laboratory (CRL)/Japan.
- Danish Space Research Institute (DSRI)/Denmark.
- European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
- European Telecommunications Satellite Organization (EUTELSAT)/Europe.
- Federal Service of Scientific, Technical & Cultural Affairs (FSST&CA)/Belgium.
- Hellenic National Space Committee (HNSC)/Greece.
- Indian Space Research Organization (ISRO)/India.
- Institute of Space and Astronautical Science (ISAS)/Japan.
- Institute of Space Research (IKI)/Russian Federation.
- KFKI Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- MIKOMTEK: CSIR (CSIR)/Republic of South Africa.
- Korea Aerospace Research Institute (KARI)/Korea.
- Ministry of Communications (MOC)/Israel.
- National Oceanic & Atmospheric Administration (NOAA)/USA.
- National Space Program Office (NSPO)/Taipei.
- Swedish Space Corporation (SSC)/Sweden.
- United States Geological Survey (USGS)/USA.

DOCUMENT CONTROL

Document	Title	Date	Status
CCSDS 103.0-B-1	Packet Telemetry Services, Issue 1	May 1996	Original Issue (supeseded)
CCSDS 103.0-B-2	Packet Telemetry Service Specification, Issue 2	June 2001	Current Issue: adds specifications to support use of the CCSDS Version-1 Telemetry Transfer Frame to transport other types of packets in addition to CCSDS Version-1 Packets, including CCSDS Network Protocol (NP) Packets and Internet Protocol (IP) packets.

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

CONTENTS

CONTENTS (continued)

Section Page

Figure

Table

1 INTRODUCTION

1.1 PURPOSE

The purpose of this Recommendation is to define the services of a packet telemetry system. To do so, it establishes a layered model of Packet Telemetry protocols and defines Packet Telemetry Services by specifying the behavior at the service interfaces to each layer. The layered model and services are based on the CCSDS Recommendations for *Packet Telemetry* and *Telemetry Channel Coding*, references [1] and [2]. These referenced Recommendations define the formats of the protocol-data-units used to transfer telemetry from spacecraft to ground or spacecraft to spacecraft, as well as the protocol procedures that support that transfer.

NOTE – Definitions of 'service', 'layer', and other terms used in this Recommendation are provided in 1.6, and are further explained in 2.2.

1.2 SCOPE

This Recommendation defines only the services provided between protocol layers of the CCSDS space to ground link. It does not define the extension of these services across distributed components of the spacecraft or ground data systems.

1.3 APPLICABILITY

This Recommendation applies to the creation of Agency standards and to the future exchange of Packet Telemetry between CCSDS Agencies in cross-support situations. The Recommendation includes comprehensive specification of the services that can be provided by remote space vehicles (spacecraft) for telemetering to space mission data processing facilities (which are usually located on Earth). The Recommendation does not attempt to define the architecture or configuration of these data processing facilities, except to describe assumed data processing services which affect the selection of certain on-board formatting options.

1.4 RATIONALE

The rationale for Packet Telemetry is presented in reference [B2].

1.5 DOCUMENT STRUCTURE

The remainder of this document is organized as follows:

- section 2 provides an overview of this Recommendation, including a layered model of packet telemetry services;

- section 3 describes the data that is transferred from data sources in space to data sinks on the ground by the packet Telemetry Services;

- section 4 defines Space Transfer services, which support the transfer of data units (created by applications) by means of Virtual Channels;
- section 5 defines Virtual Channel Access services, which provide for the transfer of Virtual Channel, and certain application data units, in a single stream of fixed-length Transfer Frames;
- section 6 defines Channel Access Coding services, which support the transfer of a stream of Transfer Frames over a noisy channel;
- annex A lists acronyms and abbreviations used in this text along with their definitions;
- annex B lists informative references;
- annex C provides a brief tutorial on OSI service terminology.

1.6 CONVENTIONS AND DEFINITIONS

1.6.1 DEFINITIONS FROM REFERENCED DOCUMENTS

The definitions below were adapted from references [B3], [B4], and [B5]. Concepts related to these terms are discussed in section 2 and in annex C of this Recommendation.

association: a cooperative relationship among entities in the same layer.

blocking: a protocol function that maps multiple service-data-units into one protocol-data-unit.

multiplexing: a protocol function that uses one association in the layer below to support more than one association for users of the protocol.

one-way communication: data communication in one pre-assigned direction.

primitive, service primitive: an abstract, atomic, implementation-independent representation of an interaction between a service-user and its service-provider.

protocol: a set of rules and formats (semantic and syntactic) which determines the communication behavior of layer entities in the performance of communication functions.

protocol-data-unit (PDU): a unit of data specified in a protocol and consisting of protocol-control-information and possibly user data.

segmentation: a protocol function that maps one service-data-unit into multiple PDUs.

service: a capability of a layer, and the layers beneath it (a service-provider), which is provided to service-users at the boundary between the service-provider and the service-users.

NOTE – The service defines the external behavior of the service-provider, independent of the mechanisms used to provide that behavior. Layers, layer entities, application-service-elements, etc. are components of a service-provider.

service-access-point (SAP): the point at which services are provided by an entity in a layer to an entity in the layer above.

service-data-unit (SDU): an amount of information whose identity is preserved when transferred between peer entities in a given layer and which is not interpreted by the supporting entities in that layer.

service-provider: an abstract representation of the totality of those entities which provide a service to service-users; i.e., a layer, and the layers beneath it.

service-user: an entity in a single system that makes use of a service.

NOTE – The service-user makes use of the service through a collection of service primitives defined for the service.

sink: an entity that receives SDUs from a service provider.

source: an entity that sends SDUs, using a service provider.

symmetric service: in a symmetric service, the local views at the service interfaces in two systems are the same. See annex C and reference [B5] for further discussion.

unconfirmed service: in an unconfirmed service, the sending end does not receive confirmation that data that it sends has reached the receiving end.

1.6.2 TERMS DEFINED IN THIS RECOMMENDATION

The terms defined below are used throughout this Recommendation. Many other terms that pertain to specific services are defined in the appropriate sections.

aperiodic: not occurring at a *constant rate* (see below).

asynchronous: not *synchronous* (see below).

constant rate; periodic: a sequence of events in which each event occurs at a fixed time interval (within specified tolerance) after the previous event in the sequence.

synchronous: a sequence of events occurring in a fixed time relationship (within specified tolerance) to another sequence of events. Note that 'synchronous' does not necessarily imply 'constant rate'.

user-optional: a qualification of a service capability indicating that the entity using the service may choose to use, or not to use, the capability. The service provider is presumed to provide the capability if requested, but also to be able to provide service that does not include the user-optional capability.

NOTE – An example of a user-optional capability is a Data-Quality Flag that a receiving user may choose not to receive.

1.6.3 USE OF BOLDFACE

Boldface characters are used for names of Packet Telemetry data units, layers and services.

1.7 REFERENCES

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

[1] *Packet Telemetry*. Recommendation for Space Data System Standards, CCSDS 102.0-B-5. Blue Book. Issue 5. Washington, D.C.: CCSDS, November 2000.

[2] *Telemetry Channel Coding*. Recommendation for Space Data System Standards, CCSDS 101.0-B-5. Blue Book. Issue 5. Washington, D.C.: CCSDS, June 2001.

[3] *CCSDS Global Spacecraft Identification Field Code Assignment Control Procedures*. Recommendation for Space Data System Standards, CCSDS 320.0-B-2. Blue Book. Issue 2. Washington, D.C.: CCSDS, October 1998.

[4] *Space Link Identifiers*. Draft Recommendation for Space Data System Standards, CCSDS 135.0-R-1. Red Book. Issue 1. Washington, D.C.: CCSDS, November 2000.

2 OVERVIEW

2.1 PACKET TELEMETRY SERVICES

This Recommendation complements the CCSDS Recommendations for *Packet Telemetry* (reference [1]), and *Telemetry Channel Coding* (reference [2]). The Packet Telemetry and Telemetry Channel Coding Recommendations

- define data units for telemetry systems;
- define formats of these data units;
- define rules and procedures for creation and use of these data units.

These Recommendations, however, do not specify the interface between a data source or sink and the entity providing transfer of data units from space to ground, nor do they explicitly define the characteristics of the transfer process, from the viewpoint of a data source or sink. This Recommendation for Packet Telemetry Services

- defines a layered model of a packet telemetry system consistent with references [1] and [2];
- defines services provided by each layer;
- provides parameters and conditions for use of each service.

This Recommendation *does not* alter or redefine reference [1] or [2].

2.2 RELATIONSHIP TO OSI

This Recommendation defines Packet Telemetry Services in the style established by the OSI Basic Reference Model (reference [B3]), which describes communications services as being provided by layers of protocols, each layer providing a service interface to users of the service in the layer above, as shown in figure 2-1. The concepts and terminology of the OSI Basic Reference Model are summarized in annex C.

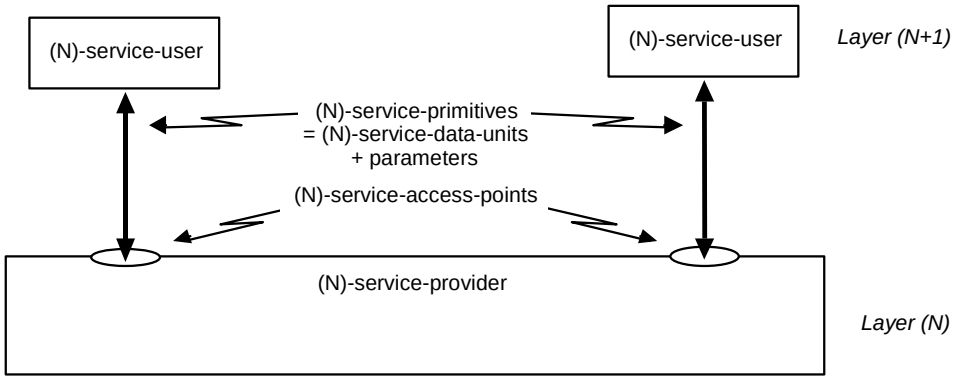


Figure 2-1: OSI Service Concept

A service interface is defined in terms of 'primitives', which present an abstract model of the exchange of data structures and control information between the layer and the service user. The primitives are independent of specific implementation approaches, and so do not specify aspects of the service interface that might vary from one implementation to another. These local issues include handshaking and flow control across the service interface (i.e., between the service user, in one layer, and the protocol entity in the layer below).

A 'service user' is not a mission user, such as a scientific investigator or spacecraft operator. A service user is typically a software process that is part of an instrument, subsystem, or data handling system on a spacecraft, or part of a data capture or data processing system on the ground.

2.3 PACKET TELEMETRY LAYERS

Although this Recommendation uses OSI concepts to define services, the services of Packet Telemetry are not structured into the same seven layers as are OSI services. Further, because a key design goal of Packet Telemetry is efficient use of limited space link resources, the Packet Telemetry PDUs are structured differently from those of OSI protocols. Because of these differences, the Packet Telemetry layers are identified by letters (A through D) rather than numbers (1 to 7), to avoid confusion with the seven OSI layers. Figure 2-2 illustrates the Packet Telemetry layers, and table 2-1 summarizes the services that these layers provide.

The services specified in this Recommendation are unidirectional services: one end (the spacecraft) can send, but not receive, data through the protocols providing the service, while the other end (on the ground) can receive, but not send.

These services are also *unconfirmed* services: the sending end (spacecraft) does not receive confirmation that data it sends has been received. This is a consequence of the design of the space link protocols, which avoid the delays involved in confirmed services.

These services can be implemented as asymmetrical services, in which the local view in one system is not the same as that in another system. That is, the implementation of the layers in one set of subsystems in space need not be structured in the same way as another set of subsystems on the ground.

Table 2-1: Summary of Packet Telemetry Services

Layer	Services	Service Capabilities
D Space Transfer layer	i. Packet Transfer Service	Transfer of a sequence of variable-length PACKETs from a source application in space to one or more sink applications on the ground.
	ii. Privately Defined Data Service	Transfer of a sequence of PRIVATELY DEFINED DATA units of fixed length, along with status fields, from an on-board source to data sinks on the ground.
	iii. Virtual Channel Frame Secondary Header Service	Synchronous transfer of fixed-length FRAME SECONDARY HEADER in each frame on the VIRTUAL CHANNEL.
	iv. Virtual Channel Operational Control Field Service	Synchronous transfer of a fixed-length OPERATIONAL CONTROL FIELD in each frame of the VIRTUAL CHANNEL.
C Virtual Channel Access layer	i. Virtual Channel Frame Service	Transfer of Transfer FRAMES from each of one to eight VIRTUAL CHANNELS over one MASTER CHANNEL.
	ii. Master Channel Frame Secondary Header Service	Synchronous transfer of a fixed-length FRAME SECONDARY HEADER in each frame on a MASTER CHANNEL.
	iii. Master Channel Operational Control Field Service	Synchronous transfer of a fixed-length OPERATIONAL CONTROL FIELD in each frame of the MASTER CHANNEL.
B Channel Access layer	Channel Access Service	Constant-rate transfer of fixed-length TRANSFER FRAMES, with optional error detection/correction.
A Physical Access layer	Physical Access Service	Provision of a modulated radio link from spacecraft to ground. This service is not within the scope of this Recommendation, but is shown in this model since it provides the foundation for services defined here.

The emphasis in this Recommendation is on descriptions of a single instance of a type of service. It does not treat system engineering issues (such as relationships among various users of a particular data transfer service, or among those system elements that provide these services). Such issues are discussed in the Packet Telemetry Concepts and Rationale Report (reference [B2]).

This Recommendation makes no assumptions concerning the allocation of services, or the functions that provide services, to particular systems, subsystems or components, either on board a spacecraft, or in a ground system. Thus this Recommendation provides a design-independent description of services that could be provided, reserving for each mission the choice of which services to implement, and how to do so.

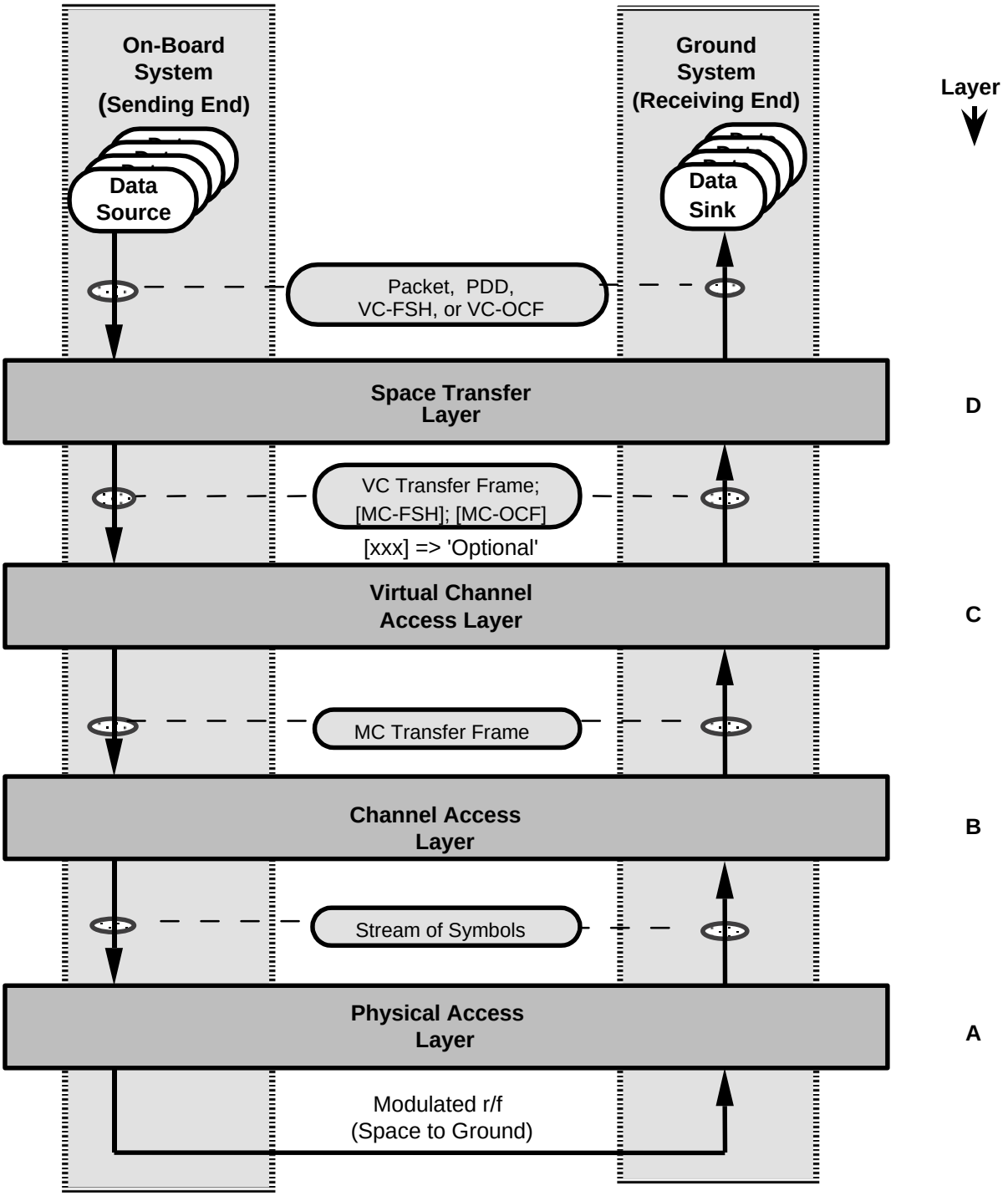


Figure 2-2: CCSDS Packet Telemetry Layers

2.4 QUEUED VERSUS BUFFERED SERVICE

Packet Telemetry Services are of two types: queued and buffered.

Queued service—In queued service (figure 2-3), each SDU from a sending user is placed in a queue, the contents of which are sent to a receiving user (or users) 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 distinctive feature of queued service is that all of the data units from the sending user are transferred, and transferred only once.

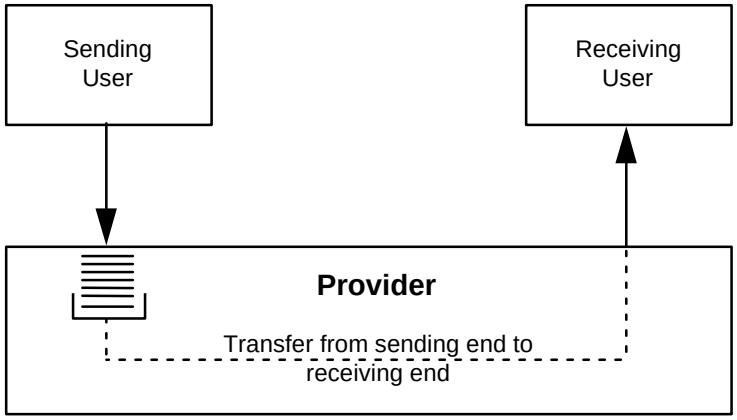


Figure 2-3: Queued Service Model

Buffered service—In buffered service (figure 2-4), each SDU from a sending user is placed in a buffer that can hold only one SDU; the contents of the buffer are sent to a receiving user (or users) at a time determined by the service (but usually known to the user). The timing may be constant rate (e.g., in every Transfer Frame sent by a spacecraft), or aperiodic (e.g., in every Transfer Frame of a Virtual Channel that produces frames at irregular intervals depending on the arrival of packets). The distinctive feature of buffered service, which is essentially time-division multiplexing, is that the timing of data transfer is driven by the service provider, not by the user. Thus a particular 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).

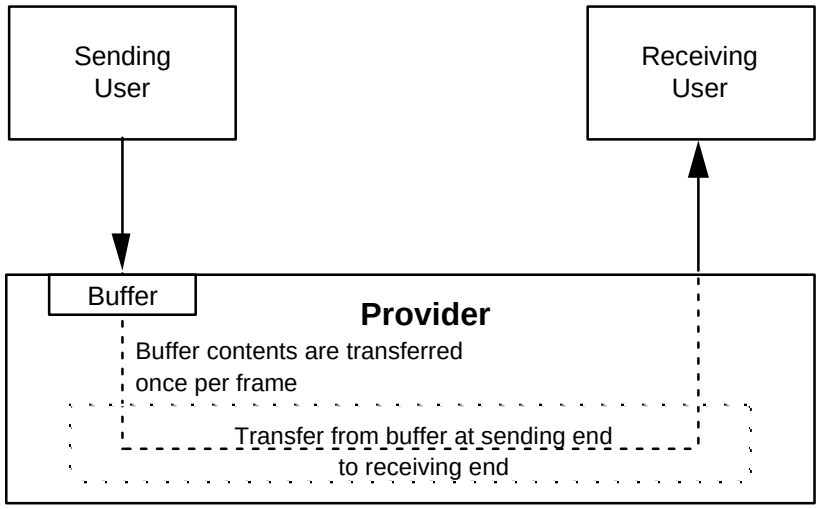


Figure 2-4: Buffered Service Model

These models of queued and buffered service are intended only to illustrate the characteristics of services. They are not intended to guide or restrict design of on-board or ground systems.

3 SOURCE DATA

3.1 SOURCE DATA OVERVIEW

3.1.1 This section describes the data that is transferred from data sources in space to data sinks on the ground by the Packet Telemetry Services described in sections 4 through 6.

3.1.2 The on-board data sources format the data units according to the specifications for the data units defined in reference [1]. These data units are

a) **Packet**;

b) the **Privately Defined Data (PDD)** field;

c) the **Frame Secondary Header (FSH)**;

d) the **Operational Control Field (OCF)**.

3.1.3 These data units are passed to the **Space Transfer layer** for transfer across the space/ground link. On the ground are the sinks that accept the transferred data units.

3.1.4 The purpose of this subsection is to establish the requirements for formatted data units produced by on-board data sources, so that the interface requirements for the lower-layer services can be met. These service definitions also identify the data units delivered to sink applications by each of the transfer services provided by lower layers.

3.2 PACKET DATA

Packets are variable-length, delimited, octet-aligned data units, and are usually the protocol data unit of a Network Layer protocol. Packets are transferred over a space link with the Packet Service. The Packets transferred by this service must be assigned a Packet Version Number (PVN) by CCSDS. For the Packet Version Numbers presently authorized by CCSDS, see reference [4]. 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.

3.3 PRIVATELY DEFINED DATA

3.3.1 **PDD** units are fixed-length data units from source applications on board that can be transferred to sink applications on the ground. A **PDD** unit consists of an integral number of octets, the format of which is not defined by CCSDS.

3.3.2 Along with each **PDD** unit, **PDD Status Field**s are provided by the on-board source for transfer to the ground. The **PDD Status Field**s are the CCSDS **Transfer Frame First Header Pointer Field** and three other bits of the transfer frame Status Field: the **Packet Order Flag** (1 bit), and **Segment Length ID** (2 bits). These are undefined by CCSDS when a **Virtual Channel** is used to transfer **PDD**. They may (optionally) be used to convey information on the validity, sequence, or other status of the data in the **PDD**. Provision of this field is mandatory; semantics are user-optional.

3.3.3 Use of **PDD** requires the specification of managed parameters, which serve to establish associations between space and ground protocol entities, specify address mappings, provide access authorization, and define operating limits.

3.3.4 Managed parameters include

a) addressing information needed to route **PDD** units to the **Virtual Channel** which is to provide the underlying **PDD Transfer Service**;

b) the fixed length of the **PDD** units (see reference [1], section 5);

c) implementation-specific parameters for timing, latency, or flow control.

3.3.5 Neither this Recommendation nor reference [1] specifies methods, procedures, or formats for providing these managed parameters.

3.3.6 **PDD** units are transferred by the **PDD Transfer** (**PDD-XFR**) **Service** to data sinks on the ground (see 4.3).

3.4 FRAME SECONDARY HEADER DATA

3.4.1 The **FSH** reference [1], section 5, is a data structure that carries fixed-length data from a source on board to a sink on the ground. Except for the **FSH** header defined in reference [1], CCSDS specifies no format or semantics for the content of an **FSH**.

3.4.2 **FSH**es may be sent in every frame of a **Virtual Channel**, using **Virtual Channel FSH** (**VC_FSH**) **Transfer** (**VC_FSH-XFR**) **Service** (see 4.4), or in every frame of a **Master Channel**, using **Master Channel FSH** (**MC_FSH) Transfer** (**MC_FSH-XFR**) **Service** (see 5.3).

3.4.3 Since **MC_FSH-XFR** and **VC_FSH-XFR** are buffered services as defined in 2.4, the creation and formatting of data to be transferred in **FSH**es may or may not be synchronized with the **Virtual Channel** or **Master Channel** that will provide the transfer service. Such synchronization, if required for timing or other purposes, is a mission-design issue.

3.4.4 The use of **FSH**es requires the specification of managed parameters, which serve to establish associations between space and ground protocol entities, specify address mappings, provide access authorization, and define operating limits.

3.4.5 Managed parameters include

a) addressing information needed to route the **FSH** to the **Virtual Channel** or **Master Channel** which is to provide the underlying transfer service;

b) the fixed length of the **FSH**;

c) the **FSH Version Number**;

e) implementation-specific parameters for timing, latency, or flow control.

3.4.6 Neither this Recommendation nor reference [1] specifies methods, procedures, or formats for providing these managed parameters.

3.5 OPERATIONAL CONTROL FIELD DATA

3.5.1 The **OCF** (reference [1], section 5) is a data structure that carries a fixed-length (32-bit) data unit from a source on board to a sink on the ground. As defined in reference [1], CCSDS specifies the use of the first bit of this field to indicate the type of data carried.

3.5.2 An instance of **OCF Service** may be carried in every frame of one **Virtual Channel**, using **Virtual Channel OCF** (**VC_OCF**) **Transfer** (**VC_OCF-XFR) Service** (see 4.5), or, in every frame of a **Master Channel**, using **Master Channel OCF** (**MC_OCF**) **Transfer** (**MC_OCF-XFR) Service** (see 5.4).

3.5.3 Since **MC_OCF-XFR** and **VC_OCF-XFR** are buffered services as defined in 2.4, the creation and formatting of data to be transferred in the OCF may or may not be synchronized with the **Virtual Channel** or **Master Channel** that will provide the transfer service. Such synchronization, if required for timing or other purposes, is a mission-design issue.

3.5.4 Use of **OCF**s requires the specification of managed parameters, which serve to establish associations between space and ground protocol entities, specify address mappings, provide access authorization, and define operating limits. Managed parameters include

a) addressing information needed to route the **OCF** to the **Virtual Channel**(s) or **Master Channel** which will provide the underlying transfer service;

b) **OCF** Type;

c) implementation-specific parameters for timing, latency, or flow control.

3.5.5 Neither this Recommendation nor reference [1] specifies methods, procedures, or formats for providing these managed parameters.

4 LAYER D—SPACE TRANSFER LAYER

4.1 SPACE TRANSFER LAYER OVERVIEW

4.1.1 The **Space Transfer layer** provides access to four transfer services from space to ground or space to space, using the **Virtual Channel Frame** (**VC_Frame**) as its PDU. These services are

- **Packet Transfer (PT-XFR) Service**;
- **PDD-XFR Service**;
- **VC_FSH-XFR Service**;
- **VC_OCF-XFR Service**.

NOTE – **PT-XFR Service** and **PDD-XFR Service** are mutually exclusive on any one **Virtual Channel** during a mission phase. A given **Virtual Channel**, whether it carries **Packet**s or **PDD**, may also carry an **FSH**, an **OCF**, both, or neither.

4.1.2 The interface between on-board users of space transfer services and the **Space Transfer layer** is illustrated in figure 4-1. This figure shows only a few of the possible combinations of services and users, and thus should not be interpreted as specification.

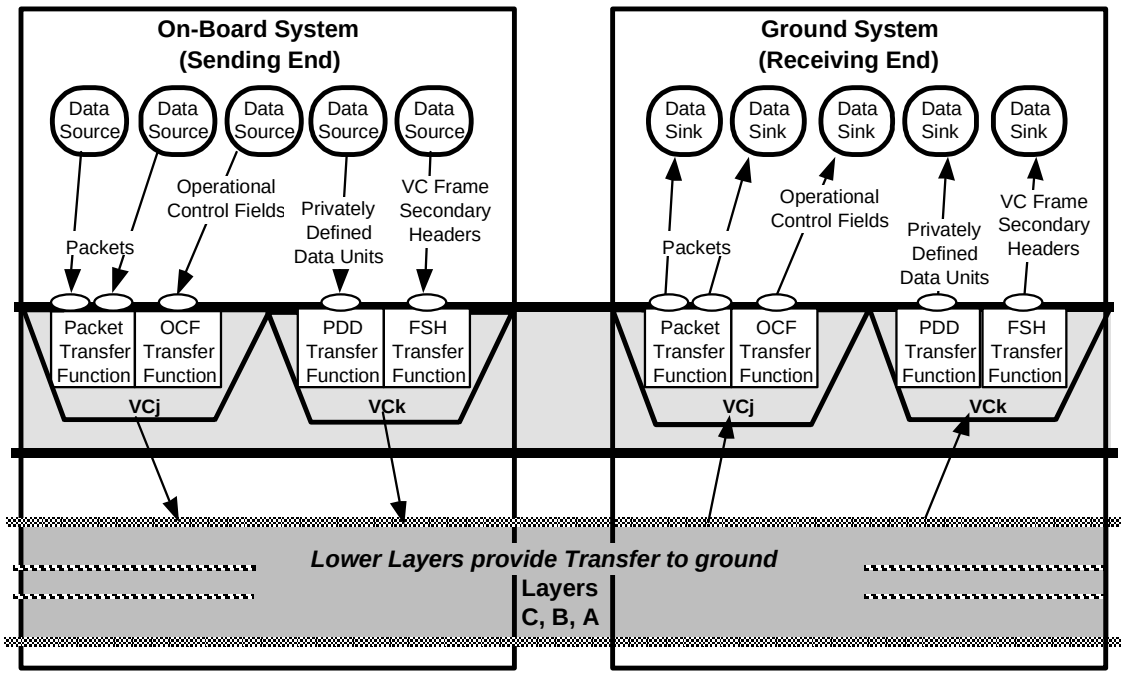


Figure 4-1: Examples of Space Transfer Layer Services

4.2 PACKET TRANSFER (PT_XFR) SERVICE

4.2.1 OVERVIEW OF PACKET TRANSFER SERVICE

The 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 Packet Version Number (PVN) authorized by CCSDS. For the PVNs presently authorized by CCSDS, see reference [4]. The service is unidirectional (one way), asynchronous and sequence preserving. It does not guarantee completeness, nor does it signal gaps in the sequence of service data units delivered to a receiving user.

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

4.2.2 PACKET SERVICE PARAMETERS

NOTE – The parameters used by the Packet Service primitives are described below.

4.2.2.1 Packet

The parameter Packet is the service data unit transferred by the Packet Service. For restrictions on the Packets transferred by the Packet Service, see reference [4].

4.2.2.2 GVCID

The GVCID is part of the SAP address of the Packet Service, and indicates the Virtual Channel through which the Packet is to be transferred. The GVCID consists of an MCID and a VCID.

4.2.2.3 Packet Version Number

The Packet Version Number is part of the SAP address of the Packet Service, and identifies the protocol entity of the upper layer that uses the Packet Service.

4.2.2.4 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.

4.2.3 PACKET SERVICE PRIMITIVES

4.2.3.1 General

The service primitives associated with this service are:

a) PACKET.request

b) PACKET.indication

The PACKET.request primitive is passed from the Packet Service user at the sending end to the service provider to request that a Packet be transferred to the user at the receiving end through the specified Virtual Channel.
The PACKET.indication is passed from the service provider to the Packet Service user at the receiving end to deliver a Packet.

4.2.3.2 PACKET.request

4.2.3.2.1 Function

The PACKET.request primitive is the service request primitive for the Packet Service.

4.2.3.2.2 Semantics

The PACKET.request primitive shall provide parameters as follows:

PACKET.request (Packet,
GVCID,
Packet Version Number)

4.2.3.2.3 When Generated

The PACKET.request primitive is passed to the service provider to request it to send the Packet.

4.2.3.2.4 Effect On Receipt

Receipt of the PACKET.request primitive causes the service provider to transfer the Packet.

4.2.3.2.5 Additional comments

The PACKET.request primitive is used to transfer Packets across the space link on the specified Virtual Channel.

4.2.3.3 PACKET.indication

4.2.3.3.1 Function

The PACKET.indication primitive is the service indication primitive for the Packet Service.

4.2.3.3.2 Semantics

The PACKET.indication primitive shall provide parameters as follows:

PACKET.indication (Packet,
GVCID,
Packet Version Number,
Packet Quality Indicator (optional))

NOTE – Packet Quality Indicator applies only to CCSDS Source Packets (see 4.2.2.4).

4.3 PRIVATELY DEFINED DATA TRANSFER SERVICE

4.3.1 OVERVIEW OF PDD-XFR SERVICE

The **PDD-XFR Service** provides transfer of a **PDD** unit of fixed length, along with status fields, from an on-board source to data sinks on the ground (see figure 4-4). The service is unidirectional (one way, space to ground), periodic, and order preserving. It does not guarantee completeness, but signals gaps. Only one instance of **PDD-XFR Service** can be provided on a **Virtual Channel**.

NOTE – **PDD-XFR Service** and **SP-XFR Service** (4.2) are mutually exclusive on any one **Virtual Channel**, within a mission phase.

4.3.2 DEFINITIONS AND ABBREVIATIONS

For the purposes of this Service Definition, the following definitions and abbreviations apply:

a) **PDD-XFR_SDU** — a fixed-length data unit, consisting of an integral number of octets, the format of which is not defined by CCSDS (see reference [1]);

b) sending **PDD-XFR user** — an on-board source of **PDD-XFR_SDU**s to be transferred;

c) receiving **PDD-XFR user** — a sink for **PDD-XFR_SDU**s on the ground; a process that receives all **PDD** units of a particular **Virtual Channel** on the ground;

d) **PDD-XFR SAP** — **service-access-point** for **PDD-XFR Service** on a particular **Virtual Channel**.

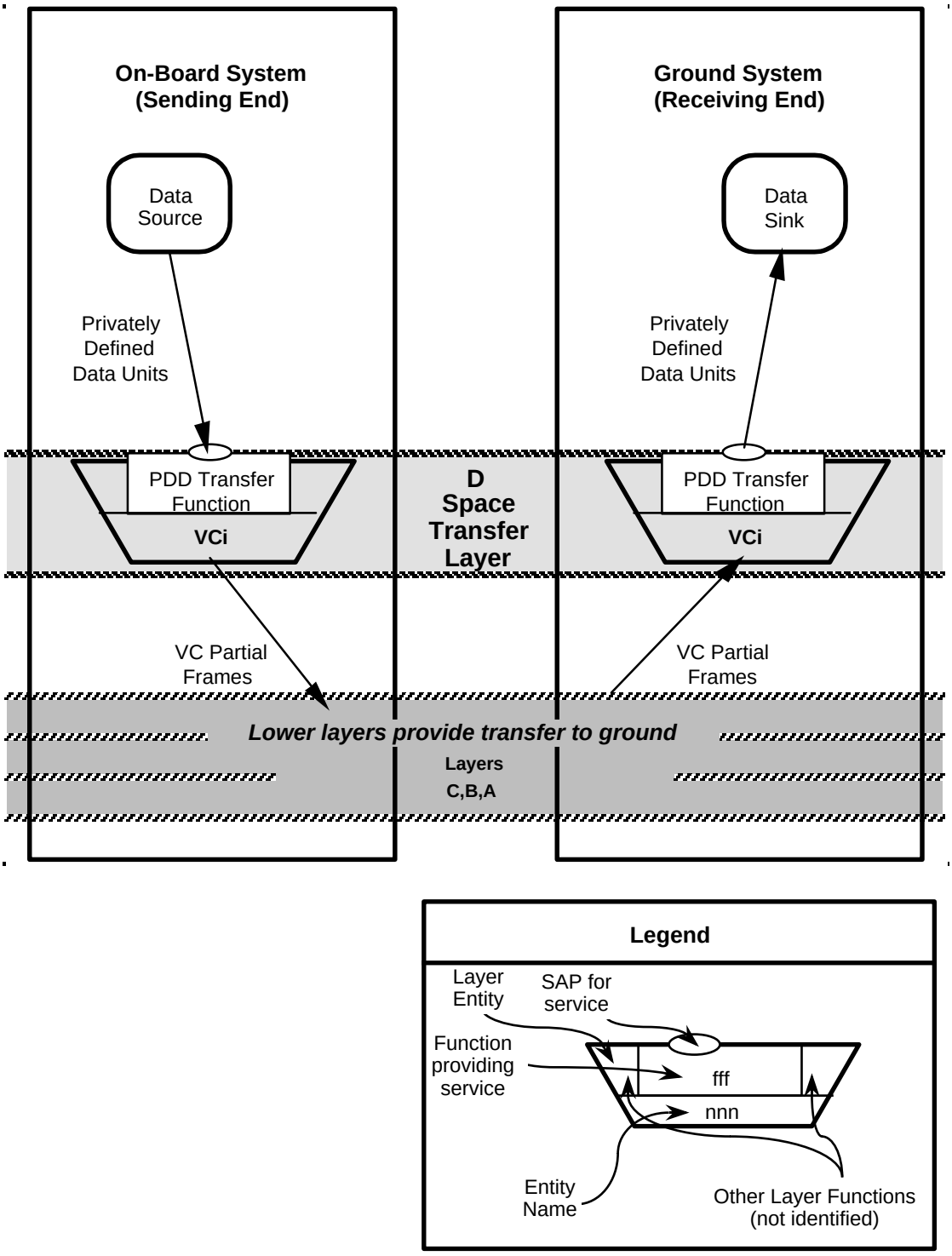


Figure 4-2: Privately Defined Data Service

4.3.3 PDD-XFR SERVICE SDU

4.3.3.1 The abstract model of **PDD-XFR Service** is a queue linking the on-board **PDD-XFR SAP** for a given **Virtual Channel** to the corresponding **PDD-XFR SAP** on the ground. A separate queue exists for each **Virtual Channel**. There shall be only one queue for **PDD-XFR Service** on a given **Virtual Channel**. This model is illustrated in figure 4-5.

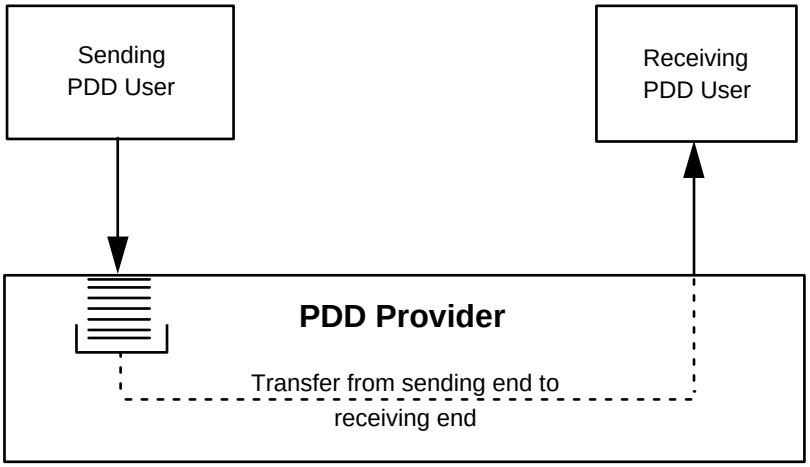


Figure 4-3: Abstract Model of PDD-XFR Service

4.3.3.2 This model implies that

a) the **PDD-XFR_SDU**s sent by an on-board source are transferred in order;

b) the time relationship between **PDD-XFR_SDU**s sent by different on-board sources, on different **Virtual Channel** s, is not specified.

4.3.4 PREREQUISITES FOR PDD-XFR SERVICE

4.3.4.1 The **PDD-XFR Service** requires **VC_Frame Service** from the layers below (see 5.2). **PDD-XFR Service** also requires the specification of managed parameters.

4.3.4.2 Managed parameters for **PDD-XFR Service** include

a) which **Virtual Channel** is to carry **PDD**;

b) which application is authorized as the source of **PDD** on the **Virtual Channel** providing the service;

c) fixed length of **Frame Data Field**;

d) routing information for delivery at receiving end;

e) whether optional parameters are to be delivered with **PDD** at the receiving end.

4.3.4.3 Neither this Recommendation nor reference [1] specifies methods, procedures, or formats for providing these managed parameters.

4.3.5 SERVICE PRIMITIVES OF THE PDD-XFR SERVICE

The service primitives associated with this service are

a) **PDD-XFR.request**

The **PDD-XFR.request** primitive is passed from the **PDD-XFR Service** user at the sending end to the **PDD-XFR SAP** of a **Virtual Channel** to request that a **PDD-XFR_SDU** be transferred.

b) **PDD-XFR.indication**

The **PDD-XFR.indication** is passed from the **Space Transfer layer** to the **PDD-XFR user** at the receiving end to deliver a **PDD-XFR_SDU**.

4.3.6 PDD-XFR SERVICE PARAMETERS

The parameters for the **PDD-XFR Service** primitives are described below.

a) **PDD-XFR_SDU** — The **PDD-XFR service-data-unit**. A **PDD-XFR_SDU** is a delimited, fixed-length data unit, consisting of an integral number of octets. The content and format of a **PDD-XFR Unit** are not further specified by the CCSDS.

b) **PDD Status Field** — The CCSDS **Transfer Frame First Header Pointer Field** and three other bits of the **Transfer Frame Status Field**: the **Packet Order Flag** (1 bit) and **Segment Length ID** (2 bits). These are undefined by CCSDS when a **Virtual Channel** is used to transfer **PDD**. They may (optionally) be used to convey information on the validity, sequence, or other status of the data in the **PDD-XFR_SDU**. Provision of this field is mandatory; semantics are user-optional.

c) **GVCID** — The **GSCID** (see reference [3]) concatenated with the **Virtual Channel Identifier** (**VCID**).

4.3.7 DETAILED PDD-XFR SERVICE SPECIFICATION

4.3.7.1 General

This subsection describes in detail the primitives and parameters associated with the **PDD-XFR Service**. The parameters are specified in an abstract sense and specify the information to be made available to the user of the primitive. The way in which a specific implementation makes this information available is not constrained by this specification.

4.3.7.2 PDD-XFR.request

a) Function:

The **PDD-XFR.request** primitive is the service request primitive for the **PDD-XFR Service**.

b) Semantics:

The **PDD-XFR.request** primitive shall provide parameters as follows:

PDD-XFR.request	**(PDD-XFR_SDU,**
	PDD Status Fields)

c) When Generated:

The **PDD-XFR.request** primitive is passed to the **Space Transfer layer** from the **PDD-XFR user** at the sending end to request that the **PDD-XFR_SDU** be transferred.

d) Effect on Receipt:

Receipt of the **PDD-XFR.request** primitive causes the **Space Transfer layer** to transfer the **PDD-XFR_SDU**.

e) Additional Comments:

1) The **PDD-XFR.request** primitive is used to transfer **PDD** units across the space link.

2) The functions performed at the sending end when the **PDD-XFR.request** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see 5.3 in *Packet Telemetry*, reference [1].

Functions include: create and number a partial **VC_Frame**; insert **PDD** into **Frame Data Field**, and **PDD Status Field** into frame header. These are components of a **Transfer Frame**, and are collectively referred to in this Recommendation as a partial **VC_Frame**.

4.3.7.3 PDD-XFR.indication

a) Function:

The **PDD-XFR.indication** primitive is the service indication primitive for the **PDD-XFR Service**.

b) Semantics:

The **PDD-XFR.indication** primitive shall provide parameters as follows:

PDD.indication	**(PDD-XFR_SDU,**
	Status Fields,
	GVCID,
	Virtual Channel Frame Count)

c) When Generated:

The **PDD-XFR.indication** primitive is passed from the **Space Transfer layer** to the **PDD Service** user to deliver a **PDD-XFR_SDU**.

d) Effect on Receipt:

The effect of receipt of the **PDD-XFR.indication** primitive by the user of the **SP-XFR** is undefined.

e) Additional Comments:

1) The **PDD-XFR.indication** primitive is used to deliver **PDD** units to the **PDD-XFR user** sink application process(es) identified by managed information at the receiving end. **Virtual Channel Frame Count** provides the means to determine if data has been lost.

2) The functions performed at the receiving end before the **PDD-XFR.indication** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see 5.3 in *Packet Telemetry*, reference [1].

Functions include: accept **VC_Frame**s from the layer below; extract **Frame Data Field**, and **PDD Status**; deliver extracted fields with **GVCID** and **Virtual Channel Frame Count**.

4.4 VIRTUAL CHANNEL FRAME SECONDARY HEADER SERVICE

4.4.1 OVERVIEW OF VC_FSH SERVICE

The **VC_FSH Service** is a unidirectional (one way, space to ground) service which provides synchronous transfer of fixed-length **FSH** in each frame on the **Virtual Channel** (see figure 4-6). The transfer is synchronized with the release of **VC_Frame**s for transfer in the **Master Channel**. The service is sequence preserving, but completeness is not guaranteed. Gaps in a sequence of **FSH**es can be detected by the receiving-end user.

NOTES

1 **VC_FSH Service** and **MC_FSH Service** are mutually exclusive.

2 Synchronization of the **VC_FSH** values to be transferred with the release of **VC_Frame**s is user-optional. It is the responsibility of each implementation (i.e., each spacecraft) to assure that the timing requirements for the **FSH** are met, and that the time of measurement of data carried in the **FSH** can be determined, if necessary.

4.4.2 DEFINITIONS AND ABBREVIATIONS

For the purposes of this Service Definition, the following definitions and abbreviations apply:

a) **VC_FSH_SDU** — an **FSH** to be sent on a particular **Virtual Channel**;

b) sending **VC_FSH user** — an on-board source of **VC_FSH_SDU**s to be transferred;

c) receiving **VC_FSH user** — a sink for **VC_FSH_SDU**s on the ground; a process that receives all **VC_FSH**es of a particular **Virtual Channel** on the ground;

d) **VC_FSH SAP** — **service-access-point** for **VC_FSH Service** on a particular **Virtual Channel**.

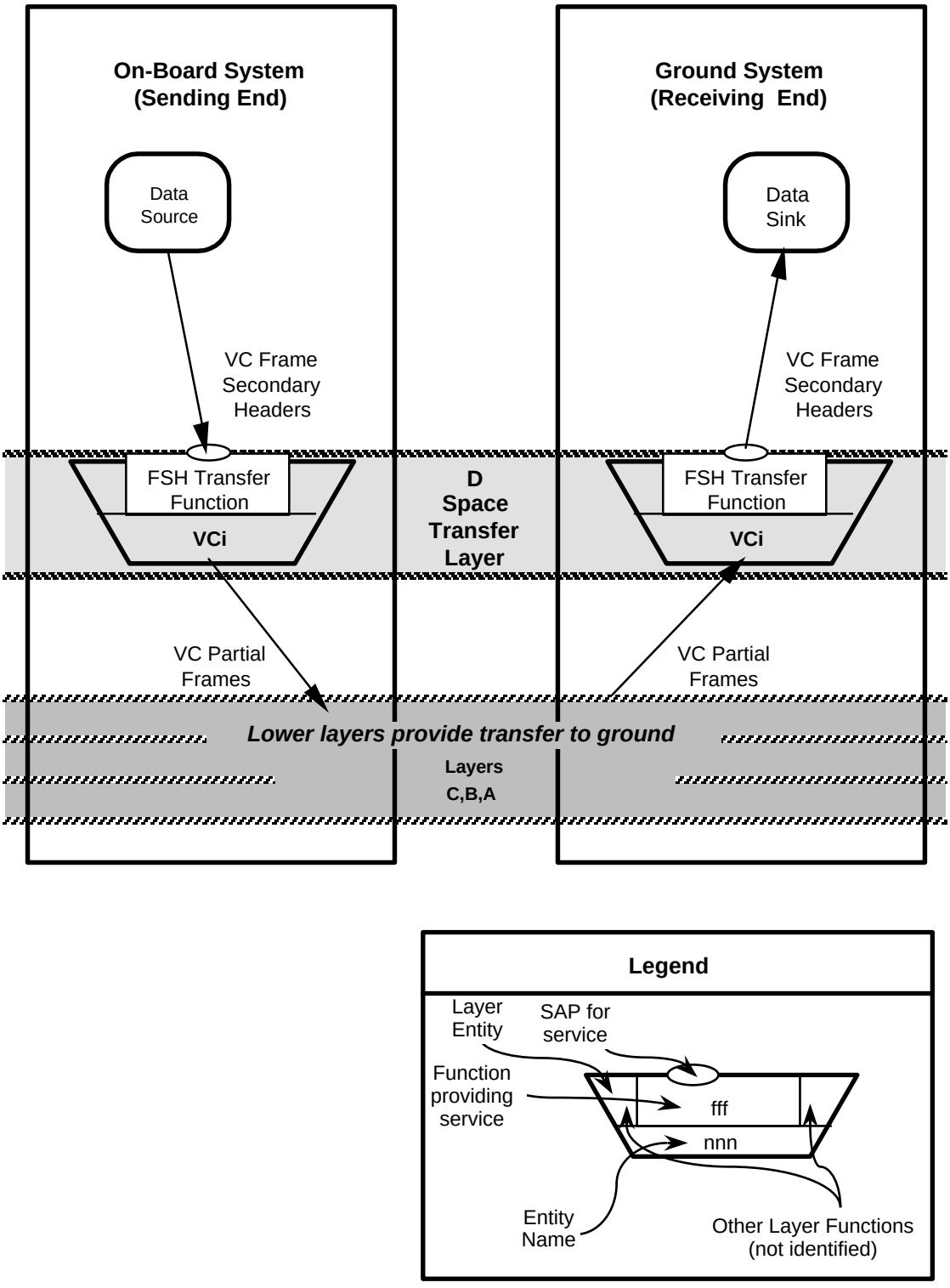


Figure 4-4: VC_Frame Secondary Header Service

4.4.3 VC_FSH SERVICE SDU

4.4.3.1 VC_FSH Service is modeled as a buffer at the sending-end **VC_FSH SAP**, the contents of which are transferred to the corresponding **VC_FSH SAP** on the ground. There shall be only one buffer for **VC_FSH Service** on a given **Virtual Channel**. This model is illustrated in figure 4-7.

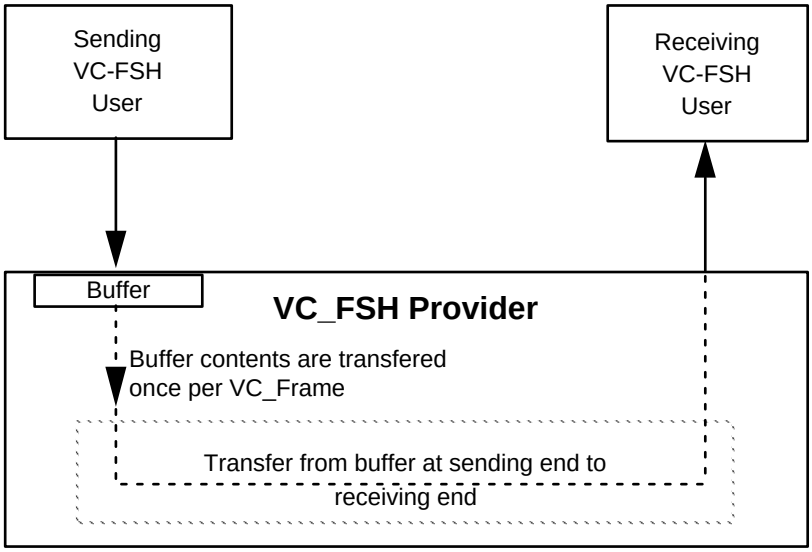


Figure 4-5: Abstract Model of VC_FSH Service

4.4.3.2 This model implies that

a) exactly one **VC_FSH_SDU** is sent per **VC_Frame**. Its value is the content of the buffer at some time between release of successive **VC_Frame**s;

b) the **VC_FSH**es sent by an on-board source are transferred in order;

c) the time relationship between placing a new value of the **VC_FSH** in the buffer, and release of a **VC_Frame** is not specified; i.e., the timing, polling, or synchronization scheme used to coordinate between **VC_FSH** source and the **VC_FSH Service** provider is mission specific.

4.4.4 PREREQUISITES FOR VC_FSH SERVICE

4.4.4.1 The **VC_FSH Service** requires **VC_Frame Service** from the layer below (see 5.2). **VC_FSH Service** also requires the specification of managed parameters, which serve to establish associations between space and ground protocol entities, specify address mappings, provide access authorization, and define operating limits.

4.4.4.2 Managed parameters for **VC_FSH Service** include

a) whether the **Virtual Channel** is to provide **VC_FSH Service**;

b) which application is authorized as the source of **VC_FSH** data on the **Virtual Channel** providing the service;

c) fixed length of the **FSH** on the **Virtual Channel**;

d) implementation-specific parameters for timing, latency, or flow control.

4.4.4.3 Neither this Recommendation nor reference [1] specifies methods, procedures, or formats for providing these managed parameters.

4.4.5 SERVICE PRIMITIVES OF THE VC_FSH SERVICE

The service primitives associated with this service are

a) **VC_FSH.request**

The **FSH.request** primitive is passed from the user of the **VC_FSH Service** at the sending end to the **VC_FSH SAP** to request that a **VC_FSH_SDU**, structured as an **FSH**, be placed into the **FSH** buffer on the specified **Virtual Channel**.

b) **VC_FSH.indication**

The **FSH.indication** is passed from the **Space Transfer layer** at the receiving end to the **VC_FSH user** to deliver an **FSH_SDU**.

4.4.6 VC_FSH SERVICE PARAMETERS

The parameters for the **VC_FSH Service** primitives are described below.

a) **VC_FSH_SDU**	The **VC_FSH service-data-unit**. An **FSH_SDU** is a fixed-length data unit consisting of an integral number of octets.
b) **Virtual Channel Frame Count**	The **Virtual Channel Frame Count** of the **Transfer Frame** carrying a **VC_FSH**.

4.4.7 DETAILED VC_FSH SERVICE SPECIFICATION

4.4.7.1 General

This subsection describes in detail the primitives and parameters associated with the **VC_FSH Service**. The parameters are specified in an abstract sense and specify the information to be made available to the user of the primitive. The way in which a specific implementation makes this information available is not constrained by this specification.

4.4.7.2 VC_FSH.request

a) Function:

The **VC_FSH.request** primitive is the service request primitive for the **VC_FSH Service**.

b) Semantics:

The **VC_FSH.request** primitive shall provide parameters as follows:

VC_FSH.request (FSH_SDU)

c) When Generated:

The **VC_FSH.request** primitive is passed to the **Virtual Channel Access** layer to request it to send the **VC_FSH_SDU**.

d) Effect on Receipt:

Receipt of the **VC_FSH.request** primitive causes the **Space Transfer layer** to transfer the **VC_FSH_SDU**.

e) Additional Comments:

1) The **VC_FSH.request** primitive is used to transfer **FSH**es.

2) The functions performed at the sending end when the **VC_FSH.request** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see 5.1.5.1 and 5.2 in *Packet Telemetry*, reference [1].

Functions include: input fixed-length **FSH**; place in buffer; synchronously insert content of buffer into **FSH** field of partial **VC_Frame**.

4.4.7.3 VC_FSH.indication

a) Function:

The **VC_FSH.indication** primitive is the service indication primitive for the **VC_FSH Service**.

b) Semantics:

The **VC_FSH.indication** primitive shall provide parameters as follows:

VC_FSH.indication (FSH_SDU,
GVCID,
Virtual Channel Frame Count)

c) When Generated:

The **VC_FSH.indication** primitive is passed from the **Space Transfer layer** to the **VC_FSH Service** user to deliver an **FSH_SDU**.

d) Effect on Receipt:

The effect of receipt of the **VC_FSH.indication** primitive by the user of the **SP-XFR** is undefined.

e) Additional Comments:

1) The **VC_FSH.indication** primitive is used to deliver **FSH**es to the **VC_FSH user** process identified by the **GVCID** (i.e., the **VCID** field in the **Transfer Frame Primary Header**, as qualified by the **GSCID**—see reference [3]). This delivery may require the use of managed information to provide routing to the **VC_FSH user** process.

2) The functions performed at the receiving end before the **VC_FSH.indication** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see 5.1.5.1 and 5.2 in *Packet Telemetry*, reference [1].

Functions include: input **VC_Frame**s; extract and deliver **VC_FSH** with **GVCID** and **Virtual Channel Frame Count**.

4.5 VIRTUAL CHANNEL OPERATIONAL CONTROL FIELD SERVICE

4.5.1 OVERVIEW OF VC_OCF SERVICE

The **VC_OCF Service** provides synchronous transfer of a fixed-length **OCF** in each frame of the **Virtual Channel** (see figure 4-8). The service is unidirectional (one way, space to ground). The transfer is synchronized with the release of **VC_Frame**s for transfer in the **Master Channel**.

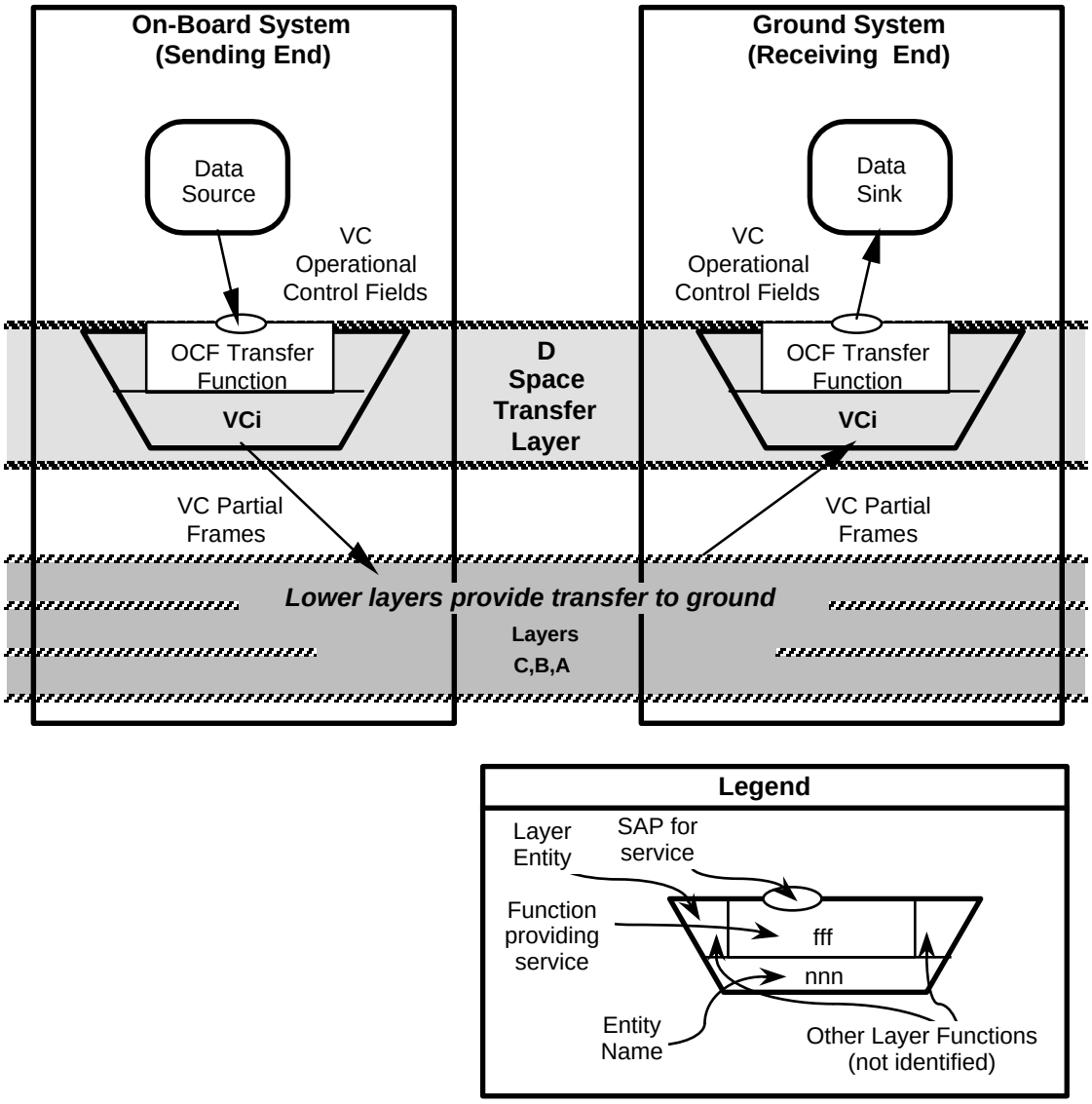


Figure 4-6: VC_OCF Service

NOTES

1 **VC_OCF Service** and **MC_OCF Service** are mutually exclusive.

2 The on-board data source providing the **VC_OCF** must make a value available for each **VC_Frame**. It is the responsibility of each implementation (i.e., each spacecraft) to assure that the timing requirements for the **VC_OCF** are met, and that the time of measurement of data carried in the **OCF** can be determined, if necessary.

4.5.2 DEFINITIONS AND ABBREVIATIONS

For the purposes of this Service Definition, the following definitions and abbreviations apply:

a) **VC_OCF_SDU** — the **VC_OCF service-data-unit**, an **OCF**;

b) sending **VC_OCF user** — an on-board source of **VC_OCF_SDU**s to be transferred;

c) receiving **VC_OCF user** — a sink for **VC_OCF_SDU**s on the ground; a process that receives the **VC_OCF_SDU**s of a particular **Virtual Channel** on the ground;

d) **VC_OCF SAP** — **service-access-point** for a **VC_OCF Service**.

4.5.3 4.5.3 VC_OCF SERVICE SDU

4.5.3.1 The abstract model of **VC_OCF Service** is a buffer at the sending-end **VC_OCF SAP**, the contents of which are transferred to the corresponding **VC_OCF SAP** on the ground. A separate buffer exists for each **Virtual Channel**. There shall be only one buffer for **VC_OCF Service** on a given **Virtual Channel**. This model is illustrated in figure 4-9.

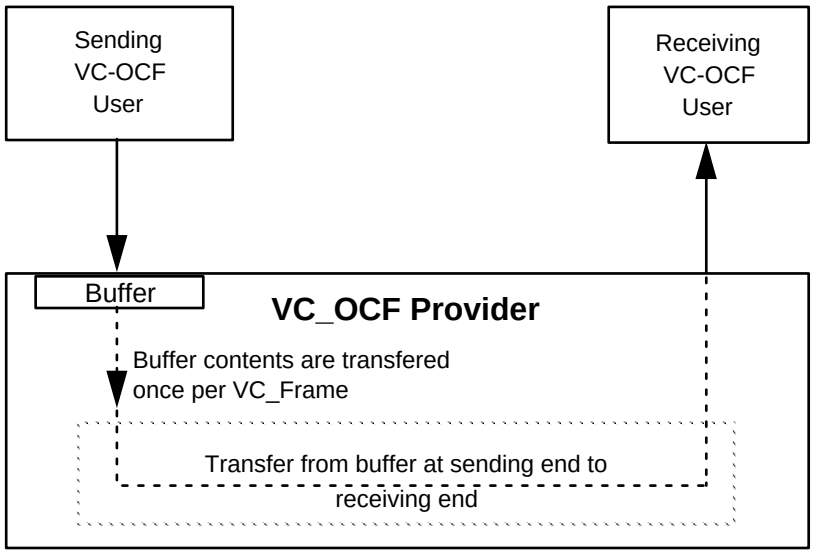


Figure 4-7: Abstract Model of VC_OCF Service

4.5.3.2 This model implies that

a) the **OCF**s sent by an on-board source are transferred in order;

b) the time relationship between **OCF**s sent by different on-board sources, on separate **Virtual Channel**s, is not specified;

c) **OCF** values are transferred only when a **VC_Frame** is released, thus some values placed in the buffer may be overwritten before they can be sent, and others may be sent more than once. Requirements for frequency or timing of **OCF** transfer are not specified by CCSDS.

4.5.4 PREREQUISITES FOR VC_OCF SERVICE

4.5.4.1 The **VC_OCF Service** requires **VC_Frame Service** from the layer below (see 5.2). **VC_OCF Service** also requires the specification of managed parameters, which serve to establish associations between space and ground protocol entities, specify address mappings, provide access authorization, and define operating limits.

4.5.4.2 Managed parameters for **VC_OCF Service** include

a) whether the **Virtual Channel** is to provide **VC_OCF Service**;

b) which application is authorized as the source of **VC_OCF** data on the **Virtual Channel** providing the service;

c) implementation-specific parameters for timing, latency, or flow control.

4.5.4.3 Neither this Recommendation nor reference [1] specifies methods, procedures, or formats for providing these managed parameters.

4.5.5 SERVICE PRIMITIVES OF THE VC_OCF SERVICE

4.5.5.1 The service primitives associated with this service are

a) **VC_OCF.request**

The **OCF.request** primitive is passed from the layer above at the sending end to the **OCF SAP** to request that a **VC_OCF_SDU** structured as an **OCF**, be inserted into the next **VC_Frame** on the specified **Virtual Channel**, and sent.

b) **VC_OCF.indication**

The **OCF.indication** is passed from the **OCF layer** at the receiving end to the **OCF user** to deliver a **VC_OCF_SDU**.

4.5.5.2 The **VC_OCF.indication** primitive is used only on the ground to deliver an **OCF** to the layer above.

4.5.6 VC_OCF SERVICE PARAMETERS

The parameters for the **VC_OCF Service** primitives are described below.

a) **VC_OCF_SDU** — The **OCF service-data-unit**. A **VC_OCF_SDU** is a fixed-length data unit consisting of four octets.

b) **VCID** — The **Virtual Channel Identifier.**

4.5.7 DETAILED VC_OCF SERVICE SPECIFICATIONS

4.5.7.1 General

This subsection describes in detail the primitives and parameters associated with the **VC_OCF Service**. The parameters are specified in an abstract sense and specify the information to be made available to the user of the primitive. The way in which a specific implementation makes this information available is not constrained by this specification.

4.5.7.2 VC_OCF.request

a) Function:

The **VC_OCF.request** primitive is the service request primitive for the **VC_OCF Service**.

b) Semantics:

The **VC_OCF.request** primitive shall provide parameters as follows:

VC_OCF.request (VC_OCF_SDU)

c) When Generated:

The **VC_OCF.request** primitive is passed to the **Virtual Channel Access** layer to request it to send the **VC_OCF_SDU**.

d) Effect on Receipt:

Receipt of the **VC_OCF.request** primitive causes the **Virtual Channel Access** layer to replace the content of the OCF buffer with the new value in the **VC_OCF_SDU**.

e) Additional Comments:

1) The **VC_OCF.request** primitive is used to transfer **OCF**s.

2) The functions performed at the sending end when the **VC_OCF.request** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see 5.1.2.3 and 5.4 in *Packet Telemetry*, reference [1].

   Functions include: input fixed-length **OCF**; synchronously insert **OCF** into **VC_Frame**.

4.5.7.3 VC_OCF.indication

a) Function:

   The **VC_OCF.indication** primitive is the service indication primitive for the **VC_OCF Service**.

b) Semantics:

   The **VC_OCF.indication** primitive shall provide parameters as follows:

   VC_OCF.indication (VC_OCF_SDU)

c) When Generated:

   The **VC_OCF.indication** primitive is passed from the **Space Transfer layer** to the **VC_OCF Service** user to deliver a **VC_OCF_SDU**.

d) Effect on Receipt:

   The effect of receipt of the **VC_OCF.indication** primitive by the user of the **SP-XFR** is undefined.

e) Additional Comments:

   1) The **VC_OCF.indication** primitive is used to deliver **OCF**s to the **VC_OCF user** process identified by the **GVCID** (i.e., the **VCID** field in the **Transfer Frame Primary Header**, as qualified by the **GSCID**—see reference [3]). This delivery may require the use of managed information to provide routing to the **VC_OCF user** process.

   2) The functions performed at the receiving end before the **VC_OCF.indication** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see 5.1.2.3 and 5.4 in *Packet Telemetry*, reference [1].

      Functions include: input **VC_Frame**s; extract and deliver **VC_OCF** with sequence quality and **Virtual Channel** sequence count (optional).

5 LAYER C—VIRTUAL CHANNEL ACCESS LAYER

5.1 OVERVIEW OF VIRTUAL CHANNEL ACCESS LAYER

Layer C provides three services to the layers above, using **Master Channel Transfer Frame**s as the supporting PDU:

a) **VC_Frame Service**;

b) **MC_FSH Service**;

c) **MC_OCF Service**.

5.2 VIRTUAL CHANNEL FRAME SERVICE

5.2.1 OVERVIEW OF VC_FRAME SERVICE

VC_Frame Service is a unidirectional service (one way, space to ground), is aperiodic, and provides transfer of **VC_Frame**s (partially formatted **Transfer Frame**s) from each of one to eight **Virtual Channel**s over one **Master Channel** (see figure 5-1). **VC_Frame**s are 'partially formatted' in that they do not contain **SCID** or **Master Channel Frame Count**, and may, or may not, contain **FSH** or **OCF**. **VC_Frame Service** is sequence preserving, but does not guarantee completeness.

5.2.2 DEFINITIONS AND ABBREVIATIONS

For the purposes of this Service Definition, the following definitions and abbreviations apply:

a) **VCF**	**VC_Frame**;
b) sending **VCF user**	an on-board source of **VC_Frame**s to be transferred; one of up to eight spacecraft **Virtual Channel**s that share the **Master Channel**;
c) receiving **VCF user**	a sink for **VC_Frame**s on the ground; a process that receives the **VC_Frame**s of a particular **Virtual Channel** on the ground;
d) **VCF SAP**	**service-access-point** for a **VCF Service**.

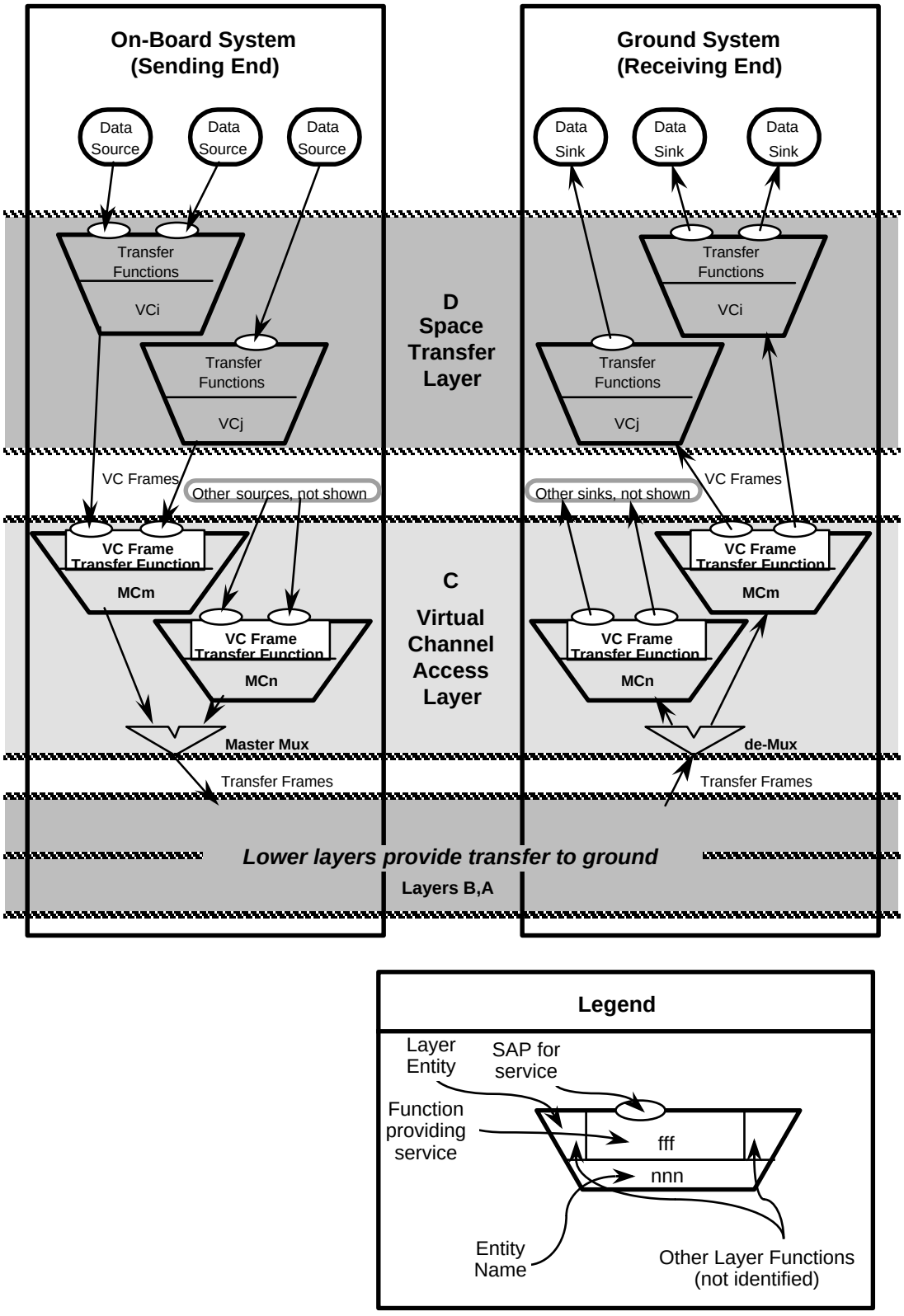


Figure 5-1: VC_Frame Service

5.2.3 VC_FRAME SERVICE SDU

5.2.3.1 The abstract model of **VC_Frame Service** is a queue linking the on-board **VCF SAP** for a given **Virtual Channel** to the corresponding **VCF SAP** on the ground. A separate queue exists for each **Virtual Channel**. This model is illustrated in figure 5-2.

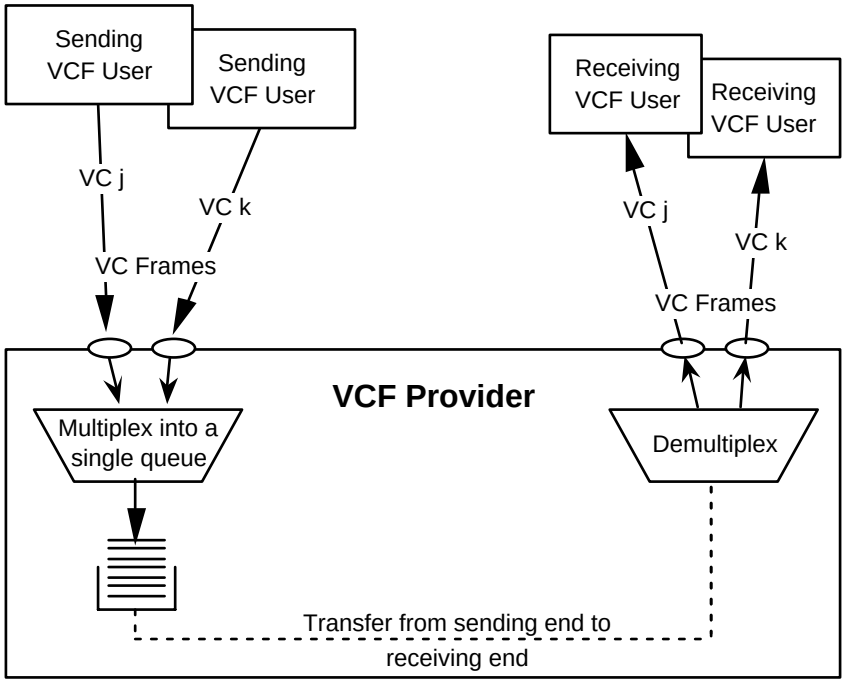


Figure 5-2: Abstract Model of a VC_Frame Service

5.2.3.2 This model implies that

a) the **VC_Frame**s sent by an on-board source are transferred in order;

b) the time relationship between **VC_Frame**s sent by different on-board sources is not specified.

5.2.4 PREREQUISITES FOR VC_FRAME SERVICE

5.2.4.1 The **VC_Frame Service** requires **Channel Access Service** from the layer below (see section 6) to provide synchronized, error-protected transmission across the space link.

5.2.4.2 The **VC_Frame Service** also requires the specification of managed parameters, which serve to establish associations between space and ground protocol entities, specify address mappings, provide access authorization, and define operating limits.

5.2.4.3 Managed parameters for **VC_Frame Service** include

a) which **Virtual Channel**s are to be provided **VC_Frame Service**;

b) frame length;

c) presence of **MC_OCF** and/or **MC_FSH** (and its length);

d) use of **Frame Error Control Field (FECF)**;

e) implementation-specific parameters for timing, latency, or flow control.

5.2.4.4 Neither this Recommendation nor reference [1] specifies methods, procedures, or formats for providing these managed parameters.

5.2.5 SERVICE PRIMITIVES OF THE VC_FRAME SERVICE

5.2.5.1 General

The service primitives associated with this service are defined below.

5.2.5.2 VCF.request

The **VCF.request** primitive is passed from the layer above at the sending end to the **VCF layer** to request that a **VCF_SDU**, structured as a **VC_Frame**, be multiplexed into the specified **Master Channel**, and sent.

5.2.5.3 VCF.indication

The **VCF.indication** is passed from the **VCF layer** at the receiving end to the **VCF user** to deliver a **VCF_SDU**.

5.2.6 VCF SERVICE PARAMETERS

The parameters for the **VCF Service** primitives are described below.

a) **VCF_SDU** — The **VCF service-data-unit**. A **VCF_SDU** is a **VC_Frame** which is a partially formatted **Transfer Frame**. A **VC_Frame** includes:

   1) all fields of the **Transfer Frame Primary Header**, excepting **Version Number**, **SCID**, **Master Channel Frame Count**, and possibly **OCF Flag** or **FSH Flag** (included only if **VC_OCF Service** or **VC_FSH Service**, respectively, are provided on the **Virtual Channel**);

2) optionally, a fixed-length **VC_FSH**;

3) a fixed-length **Transfer Frame Data Field**;

4) optionally, a fixed-length **VC_OCF**.

b) **VCID** — **Virtual Channel ID**.

c) **MC_Sequence-Quality_Indicator** — Indication provided with delivery of a **VC_Frame** (at receiving end) that there has been a gap in **Master Channel** sequence numbers since the previous **VC_Frame** was delivered on the **Virtual Channel**.

5.2.7 DETAILED VCF SERVICE SPECIFICATION

5.2.7.1 General

This subsection describes in detail the primitives and parameters associated with the **VCF Service**. The parameters are specified in an abstract sense and specify the information to be made available to the user of the primitive. The way in which a specific implementation makes this information available is not constrained by this specification.

5.2.7.2 VCF.request

a) Function:

The **VCF.request** primitive is the service request primitive for the **VCF Service**.

b) Semantics:

The **VCF.request** primitive shall provide parameters as follows:

VCF.request (VCF_SDU)

c) When Generated:

The **VCF.request** primitive is passed to the **Virtual Channel Access layer** to request it to send the **VCF_SDU**.

d) Effect on Receipt:

Receipt of the **VCF.request** primitive causes the **Virtual Channel Access layer** to transfer the **VCF_SDU**.

e) Additional Comments:

1) The **VCF.request** primitive is used to transfer **VC_Frames.**

2) The functions performed at the sending end when the **VCF.request** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see section 5, particularly subsections 5.1 and 5.5 in *Packet Telemetry*, Reference [1].

Functions include: accept **VC_Frame**s from various **Virtual Channel**s; multiplex **VC_Frame**s; add **GSCID** (see reference [3]) and **Master Channel Frame Count** to **Frame Header**; optionally, add **FECF**; output to the layer below a constant-rate stream of sequentially numbered **MC_Frame**s.

5.2.7.3 VCF.indication

a) Function:

The **VCF.indication** primitive is the service indication primitive for the **VCF Service**.

b) Semantics:

The **VCF.indication** primitive shall provide parameters as follows:

VCF.indication	**(VCF_SDU,**
	MC_Sequence-Quality_Indicator)

c) When Generated:

The **VCF.indication** primitive is passed from the **Virtual Channel Access** layer to the **VCF Service** user to deliver a **VCF_SDU**.

d) Effect on Receipt:

The effect of receipt of the **VCF.indication** primitive by the user of the **VCF Service** is defined in 4.2.7.3 e) 2), and 4.3.7.3 e) 2).

e) Additional Comments:

1) The **VCF.indication** primitive is used to deliver **VC_Frame**s to the **VCF user** process identified by the **VCID** in the **VC_Frame Header**, as qualified by the **GSCID** (see reference [3]). This delivery may require the use of managed information to provide routing to the **VCF user** processes.

2) The functions performed at the receiving end before the **VCF.indication** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see section 5, particularly 5.1 and 5.5 in *Packet Telemetry*, reference [1].

   Functions include: input a stream of numbered **MC_Frame**s; demultiplex **VC_Frame**s; optionally, check **FECF**; output **VC_Frame**s with **Master Channel** sequence quality.

5.3 MASTER CHANNEL FRAME SECONDARY HEADER SERVICE

5.3.1 OVERVIEW OF MC_FSH SERVICE

MC_FSH Service is a unidirectional service (one way, space to ground) which provides synchronous transfer of a fixed-length **FSH** in each frame on the **Master Channel** (see figure 5-3). The transfer is synchronized with the release of **MC_Frame**s. The service is sequence preserving but does not guarantee completeness.

NOTES

1 **MC_FSH Service** and **VC_FSH Service** are mutually exclusive.

2 The on-board data source providing the **FSH** must make a value available for each **Transfer Frame** on the **Master Channel**. It is the responsibility of each implementation (i.e., each spacecraft) to assure that the timing requirements for the **FSH** are met, and that the time of measurement of data carried in the **FSH** can be determined.

5.3.2 DEFINITIONS AND ABBREVIATIONS

For the purposes of this Service Definition, the following definitions and abbreviations apply:

a) **MC_FSH_SDU** — **FSH** to be sent on a **Master Channel**;

b) sending **MC_FSH user** — an on-board source of **MC_FSH_SDU**s to be transferred;

c) receiving **MC_FSH user** — a sink for **MC_FSH_SDU**s on the ground; a process that receives the **MC_FSH_SDU**s of a particular **Master Channel** on the ground;

d) **MC_FSH SAP** — **service-access-point** for a **Master Channel**.

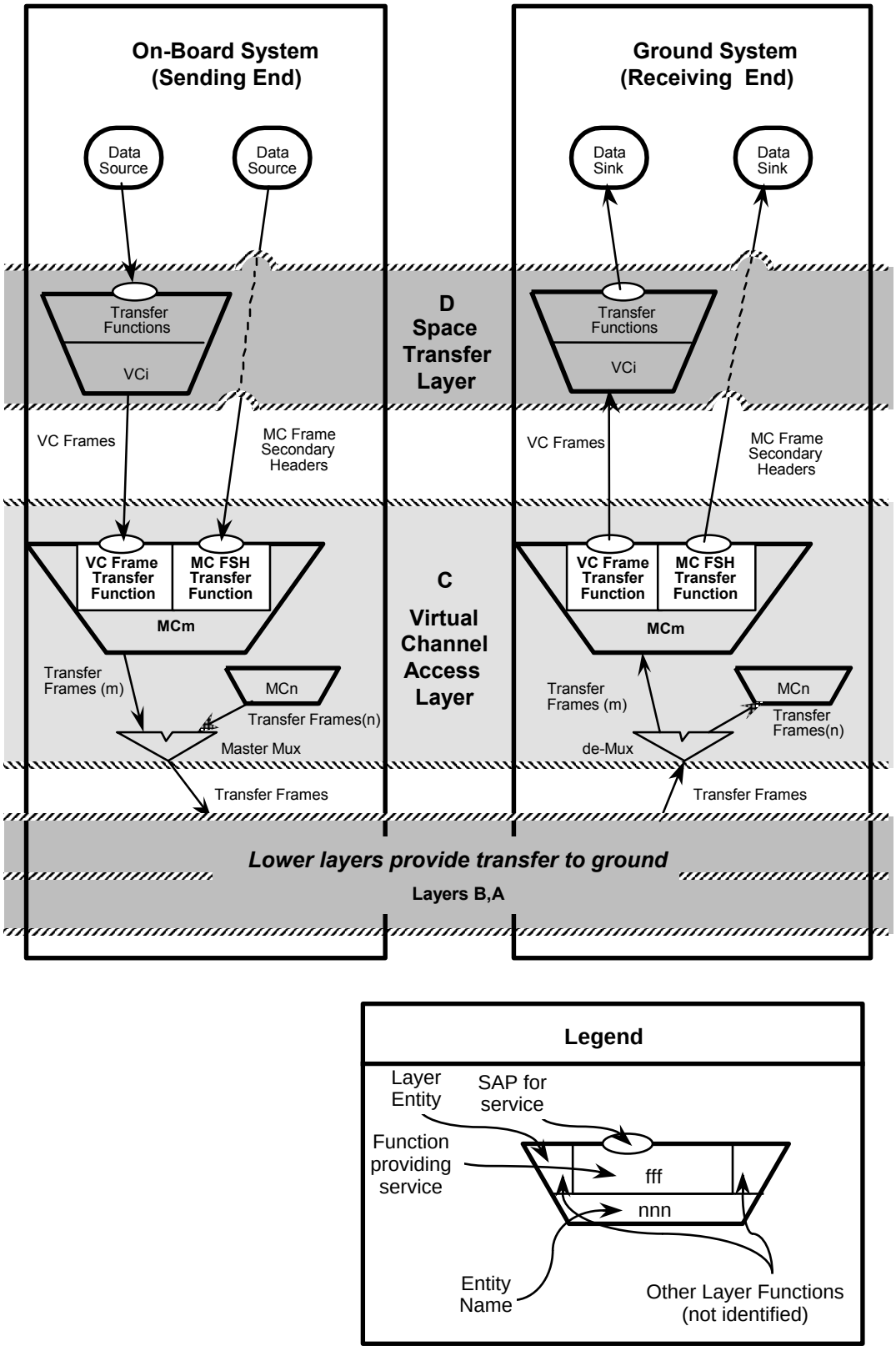


Figure 5-3: MC_Frame Secondary Header Service

5.3.3 MC_FSH SERVICE SDU

5.3.3.1 The abstract model of **MC_FSH Service** is a buffer at the sending-end **MC_FSH SAP**, the contents of which are transferred to the corresponding **MC_FSH SAP** on the ground. This model is illustrated in figure 5-4.

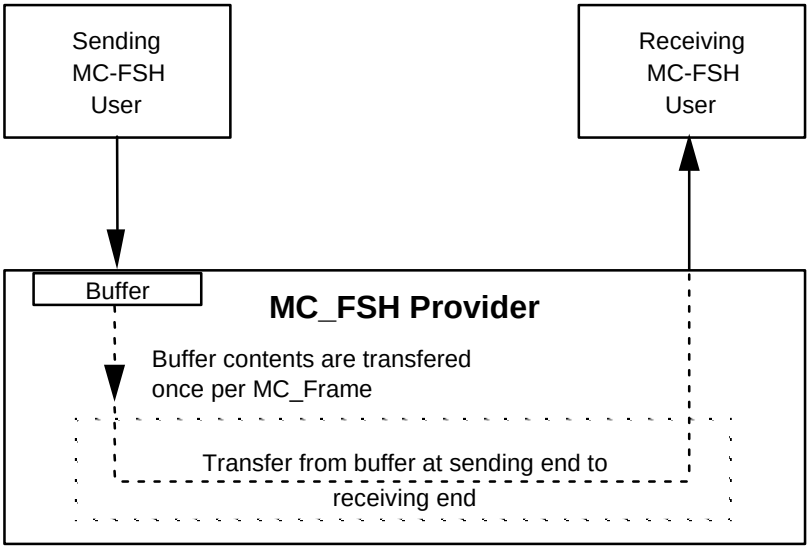


Figure 5-4: Abstract Model of an MC_FSH Service

5.3.3.2 This model implies that

a) exactly one **MC_FSH** is sent per **MC_Frame**; its value is the content of the buffer at some time between release of successive **MC_Frame**s;

b) the **MC_FSH**es sent by an on-board source are transferred in order;

c) the time relationship between placing a new value of the **MC_FSH** in the buffer, moving that value into the **FSH** field of the next **MC_Frame**, and release of that **MC_Frame**, is not specified; i.e., the timing, polling, or synchronization scheme used to coordinate between **MC_FSH** source and the **MC_FSH Service** provider is mission-specific.

5.3.4 PREREQUISITES FOR THE MC_FSH SERVICE

5.3.4.1 The **MC_FSH Service** requires **Channel Access Service** from the layer below (see section 6). **MC_FSH Service** also requires the specification of managed parameters, which serve to establish associations between space and ground protocol entities, specify address mappings, provide access authorization, and define operating limits.

5.3.4.2 Managed parameters for **MC_FSH Service** include

a) whether the **Master Channel** is to provide **MC_FSH Service**;

b) which application is authorized to be the source of **MC_FSH** data;

c) fixed length of the **FSH** on the **Master Channel**;

d) implementation-specific parameters for timing, latency, or flow control.

5.3.4.3 Neither this Recommendation nor reference [1] specifies methods, procedures, or formats for providing these managed parameters.

5.3.5 SERVICE PRIMITIVES OF THE MC_FSH SERVICE

The service primitives associated with this service are

a) **MC_FSH.request**

The **MC_FSH.request** primitive is passed from the layer above at the sending end to the **MC_FSH SAP** to request that an **MC_FSH_SDU** structured as an **FSH**, be placed in the buffer to be sent.

b) **MC_FSH.indication**

The **MC_FSH.indication** is passed from the **Virtual Channel Access** layer at the receiving end to the **MC_FSH user** to deliver an **MC_FSH_SDU**.

5.3.6 MC_FSH SERVICE PARAMETERS

The parameters for the **MC_FSH Service** primitives are described below.

a) **MC_FSH_SDU**	The **MC_FSH service-data-unit**. An **FSH_SDU** is a delimited, octet-aligned data unit which is formatted as an **FSH**.
b) **GSCID**	**Global Spacecraft Identifier** (see reference [3]). The **GSCID** identifies the **MC_FSH_SAP** for **MC_FSH_SDU**s within a specific **Master Channel**.
c) **Master Channel Frame Count**	The **Master Channel Frame Count** of the **Transfer Frame** carrying an **MC_FSH**.

5.3.7 DETAILED MC_FSH SERVICE SPECIFICATION

5.3.7.1 General

This subsection describes in detail the primitives and parameters associated with the **MC_FSH Service**. The parameters are specified in an abstract sense and specify the information to be made available to the user of the primitive. The way in which a specific implementation makes this information available is not constrained by this specification.

5.3.7.2 MC_FSH.request

a) Function:

The **MC_FSH.request** primitive is the service request primitive for the **MC_FSH Service**.

b) Semantics:

The **MC_FSH.request** primitive shall provide parameters as follows:

MC_FSH.request	**(FSH_SDU,**
	GSCID)

c) When Generated:

The **MC_FSH.request** primitive is passed to the **Virtual Channel Access layer** to request it to send the **MC_FSH_SDU**.

d) Effect on Receipt:

Receipt of the **MC_FSH.request** primitive causes the **Virtual Channel Access layer** to replace the contents of the buffer with the new value in the **MC_FSH_SDU**.

e) Additional Comments:

1) The **MC_FSH.request** primitive is used to transfer **FSH**es.

2) The functions performed at the sending end when the **MC_FSH.request** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see 5.1.5.1 and 5.2 in *Packet Telemetry*, reference [1].

Functions include: input fixed-length **FSH**; place in buffer; synchronously insert contents of buffer into **FSH** field of **MC_Frame**.

5.3.7.3 MC_FSH.indication

a) Function:

The **MC_FSH.indication** primitive is the service indication primitive for the **MC_FSH Service**.

b) Semantics:

The **MC_FSH.indication** primitive shall provide parameters as follows:

MC_FSH.indication	**(FSH_SDU,**
	GSCID,
	MASTER CHANNEL FRAME COUNT)

c) When Generated:

The **MC_FSH.indication** primitive is passed from the **Virtual Channel Access layer** to the **MC_FSH Service** user to deliver an **MC_FSH_SDU**.

d) Effect on Receipt:

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

e) Additional Comments:

1) The **MC_FSH.indication** primitive is used to deliver **FSH**es to the **MC_FSH user** process identified by the **GSCID** (i.e., the **SCID** field in the **Transfer Frame Primary Header**, as qualified by the **Version Number**—see reference [3]). This delivery may require the use of managed information to provide routing to the **MC_FSH user** process.

2) The functions performed at the receiving end before the **MC_FSH.indication** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see 5.1.5.1 and 5.2 in *Packet Telemetry*, reference [1].

Functions include: input **MC_Frame**s; extract and deliver **MC_FSH** with sequence quality and **Master Channel Frame Count**.

5.4 MASTER CHANNEL OPERATIONAL CONTROL FIELD SERVICE

5.4.1 OVERVIEW OF MC_OCF SERVICE

The **MC_OCF Service** provides synchronous transfer of a fixed-length **OCF** in each frame of the **Master Channel** (see figure 5-5). The service is unidirectional (one way, space to ground). The transfer is synchronized with the release of **MC_Frame**s. The **MC_OCF Service** is sequence preserving, but does not guarantee completeness.

NOTES

1 **MC_OCF Service** and **VC_OCF Service** are mutually exclusive.

2 The on-board data source providing the **MC_OCF** must make a value available for each **MC_Frame**. It is the responsibility of each implementation (i.e., each spacecraft) to assure that the timing requirements for the **MC_OCF** are met, and that the time of measurement of data carried in the **OCF** can be determined, if necessary.

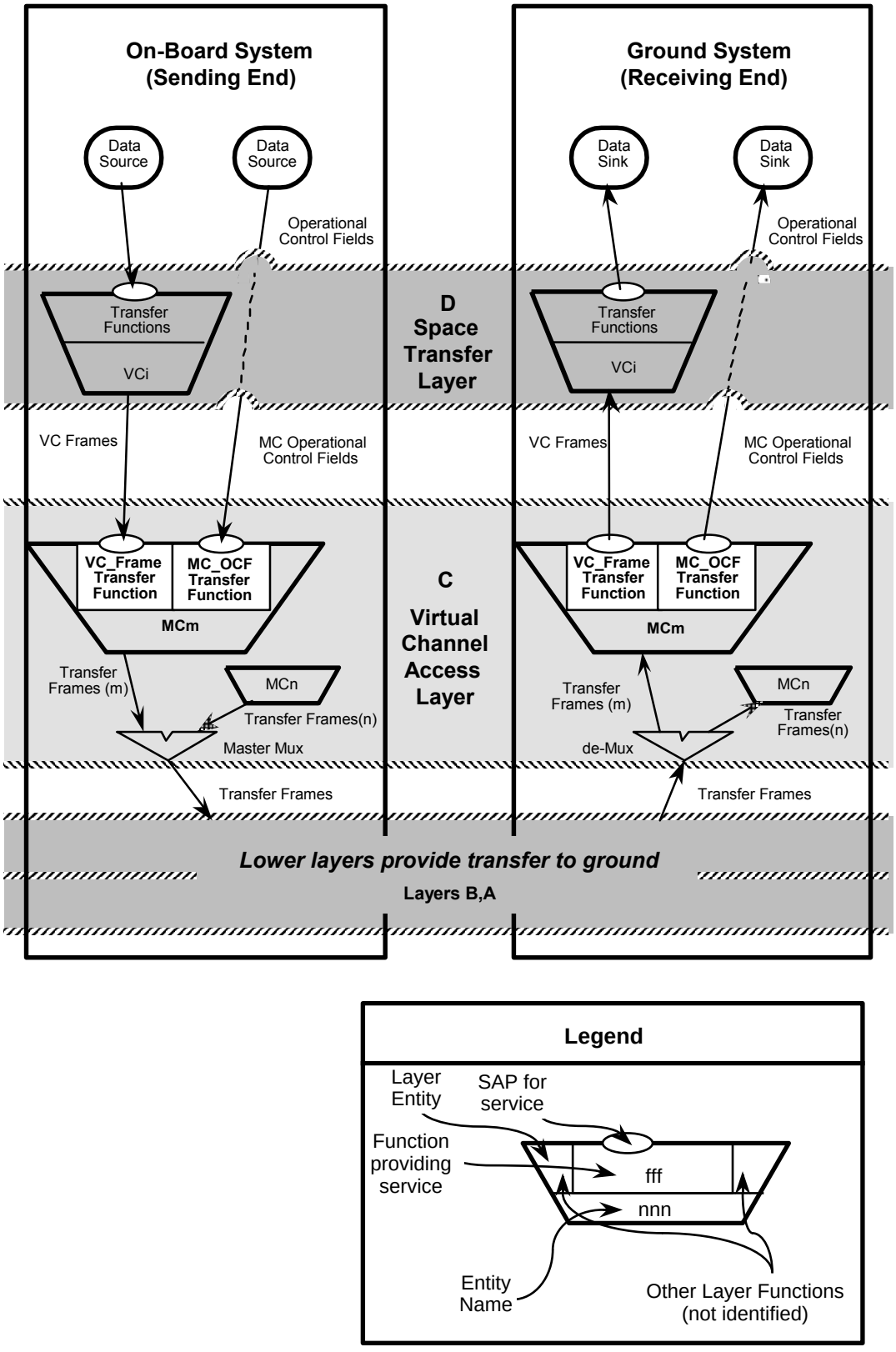


Figure 5-5: MC_OCF Service

5.4.2 DEFINITIONS AND ABBREVIATIONS

For the purposes of this Service Definition, the following definitions and abbreviations apply:

a)	**OCF**	**Operational Control Field**;
b)	sending **MC_OCF user**	an on-board source of **OCF**s to be transferred;
c)	receiving **MC_OCF user**	a sink for **OCF**s on the ground; a process that receives the **OCF**s of a particular **Master Channel** on the ground;
d)	**MC_OCF SAP**	**service-access-point** for an **MC_OCF Service.**

5.4.3 MC_OCF SERVICE SDU

5.4.3.1 The abstract model of **MC_OCF Service** is a buffer at the sending-end **MC_OCF SAP**, the contents of which are transferred to the corresponding **MC_OCF SAP** on the ground. This model is illustrated in figure 5-6.

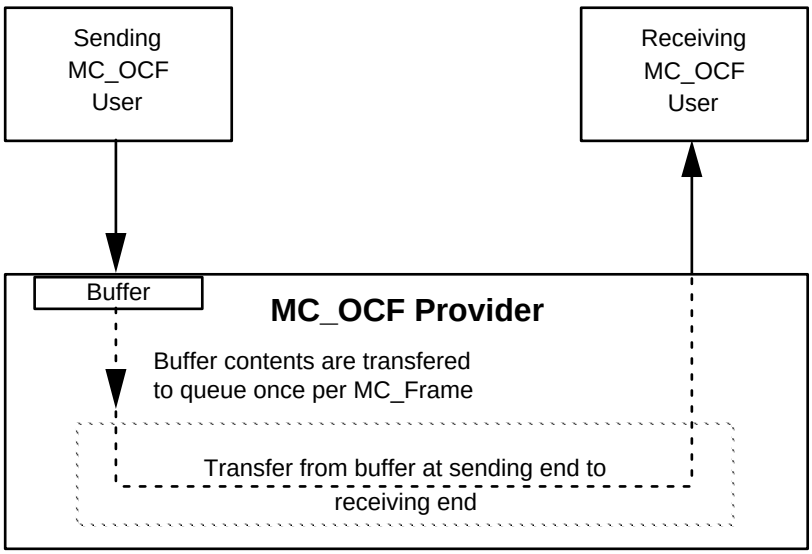


Figure 5-6: Abstract Model of MC_OCF Service

5.4.3.2 This model implies that the **OCF**s sent by an on-board source are transferred in order.

5.4.4 PREREQUISITES FOR THE MC_OCF SERVICE

5.4.4.1 The **MC_OCF Service** requires **Channel Access Service** from the layer below (see section 6). **MC_OCF Service** also requires the specification of managed parameters, which serve to establish associations between space and ground protocol entities, specify address mappings, provide access authorization, and define operating limits.

5.4.4.2 Managed parameters for **MC_OCF Service** include

a) whether the **Master Channel** is to provide **MC_OCF Service**;

b) which application is authorized as the source of **MC_OCF** data;

c) implementation-specific parameters for timing, latency, or flow control.

5.4.4.3 Neither this Recommendation nor reference [1] specifies methods, procedures, or formats for providing these managed parameters.

5.4.5 SERVICE PRIMITIVES OF THE MC_OCF SERVICE

The service primitives associated with this service are

a) **MC_OCF.request**

The **MC_OCF.request** primitive is passed from the layer above at the sending end to the **MC_OCF layer** to request that an **MC_OCF_SDU** structured as an **OCF**, be inserted into the **Transfer Frame**s of a specified **Master Channel**, and sent.

b) **MC_OCF.indication**

The **MC_OCF.indication** is passed from the **MC_OCF layer** at the receiving end to the **MC_OCF user** to deliver an **MC_OCF_SDU**.

5.4.6 MC_OCF SERVICE PARAMETERS

The parameters for the **MC_OCF Service** primitives are described below.

a) **MC_OCF_SDU**	The **MC_OCF service-data-unit**. An **MC_OCF_SDU** is a fixed-length, octet-aligned data unit.
b) **GSCID**	**Global Spacecraft Identifier** (see reference [3]).
c) **Master Channel Frame Count**	The **Master Channel Frame Count** of the **Transfer Frame** carrying an **MC_FSH**.

5.4.7 DETAILED SPECIFICATION OF MC_OCF SERVICE PRIMITIVES

5.4.7.1 General

This subsection describes in detail the primitives and parameters associated with the **MC_OCF Service**. The parameters are specified in an abstract sense and specify the information to be made available to the user of the primitive. The way in which a specific implementation makes this information available is not constrained by this specification.

5.4.7.2 MC_OCF.request

a) Function:

The **MC_OCF.request** primitive is the service request primitive for the **MC_OCF Service**.

b) Semantics:

The **MC_OCF.request** primitive shall provide parameters as follows:

MC_OCF.request	**(MC_OCF_SDU, GSCID)**

c) When Generated:

The **MC_OCF.request** primitive is passed to the **Virtual Channel Access layer** to request it to send the **MC_OCF_SDU**.

d) Effect on Receipt:

Receipt of the **MC_OCF.request** primitive causes the **Virtual Channel Access layer** to replace the content of the **OCF** buffer with the new value in the **MC_OCF_SDU**.

e) Additional Comments:

1) The **MC_OCF.request** primitive is used to transfer **OCF**s.

2) The functions performed at the sending end when the **MC_OCF.request** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see 5.1.2.3 and 5.4 in *Packet Telemetry*, reference [1].

Functions include: input fixed-length **OCF**; place in buffer; synchronously insert content of buffer into **OCF** field of **MC_Frame**.

5.4.7.3 MC_OCF.indication

a) Function:

The **MC_OCF.indication** primitive is the service indication primitive for the **MC_OCF Service**.

b) Semantics:

The **MC_OCF.indication** primitive shall provide parameters as follows:

MC_OCF.indication	**(MC_OCF_SDU,**
	GSCID,
	MASTER CHANNEL FRAME COUNT)

c) When Generated:

The **MC_OCF.indication** primitive is passed from the **Space Transfer layer** to the **MC_OCF Service** user to deliver an **MC_OCF_SDU**.

d) Effect on Receipt:

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

e) Additional Comments:

1) The **MC_OCF.indication** primitive is used to deliver **OCF**s to the **MC_OCF user** process identified by the **GSCID** (i.e., the **SCID** field in the **Transfer Frame Primary Header**, as qualified by the **Version Number**—see reference [3]). This delivery may require the use of managed information to provide routing to the **MC_OCF user** process.

2) The functions performed at the receiving end before the **MC_OCF.indication** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see 5.1.2.3 and 5.4 in *Packet Telemetry*, reference [1].

Functions include: input **MC_Frame**s; extract and deliver **MC_OCF** with sequence quality and **Master Channel** sequence count (optional).

5.5 ADDITIONAL (OPTIONAL) FUNCTIONS OF THIS LAYER

A spacecraft may produce more than one **Master Channel**, i.e., **Transfer Frame**s with more than one **GSCID** (see reference [3]). In this case, the packet telemetry system would perform the additional functions below.

a) Sending end:

   Multiplex two or more **Master Channel**s into a constant-rate stream of **Transfer Frame**s.

b) Receiving end:

   Demultiplex two or more **Master Channel**s from a constant-rate stream of **Transfer Frame**s.

6 LAYER B—CHANNEL ACCESS LAYER

6.1 OVERVIEW OF CHANNEL ACCESS SERVICE

Layer B provides **Channel Access Service**, a unidirectional service (one way, space to ground) which provides constant-rate transfer of a sequence of fixed-length **Transfer Frame**s, with optional error detection/correction (see figure 6-1).

6.2 DEFINITIONS AND ABBREVIATIONS

For the purposes of this Service Definition, the following definitions and abbreviations apply:

a) **CA_SDU** — the **Channel Access service-data-unit**; a **CA_SDU** is a fixed-length data unit consisting of an integral number of octets; there is no restriction on the bit pattern of data to be transferred; in particular, there is no requirement by this service for bit transitions within a **Transfer Frame**;

b) sending **CA user** — an on-board source of **CA_SDU**s to be transferred;

c) receiving **CA user** — a sink for **CA_SDU**s on the ground; a process that receives the **CA_SDU**s on the ground;

d) **CA SAP** — **service-access-point** for a **Channel Access Service**.

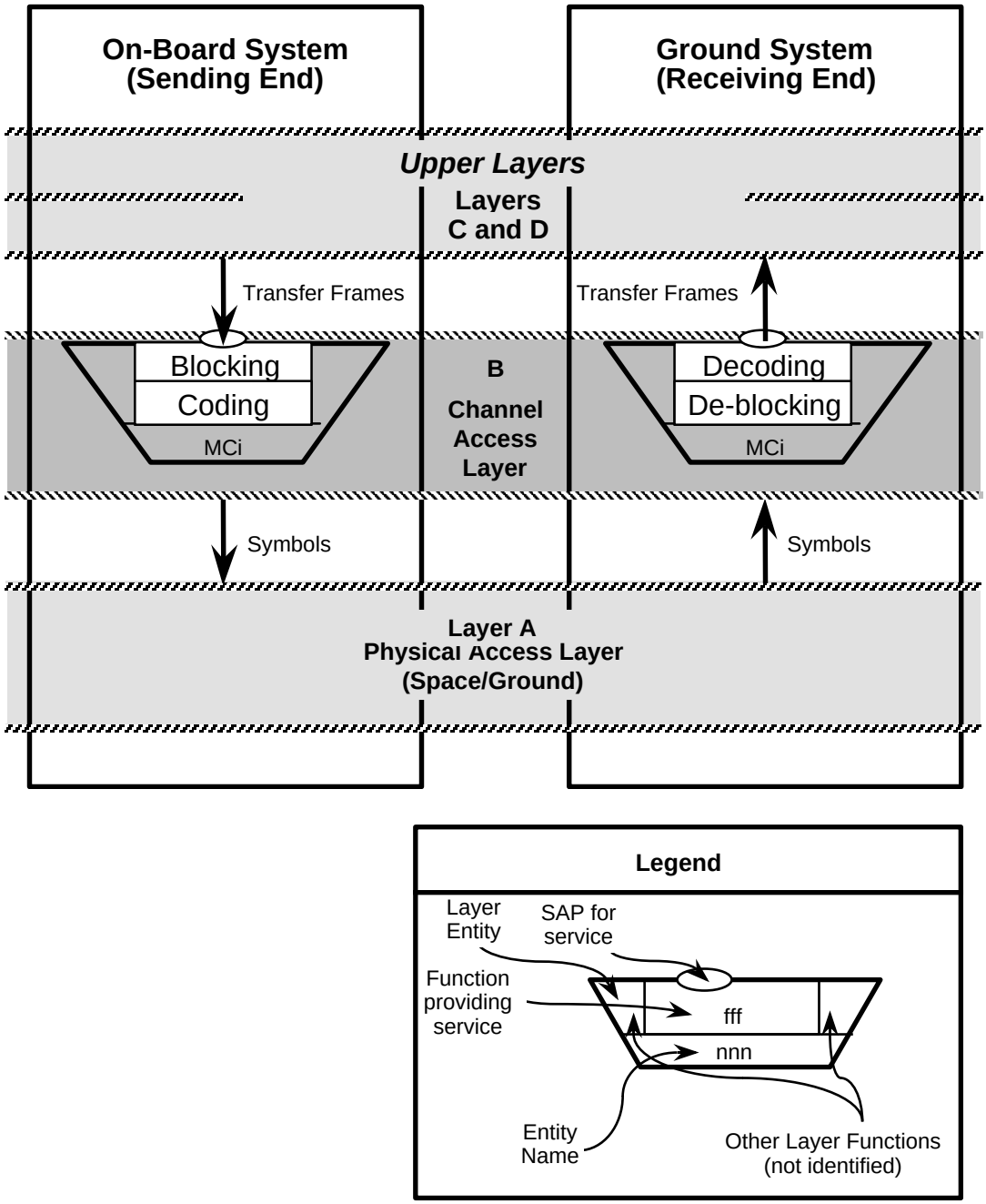


Figure 6-1: Channel Access Service

6.3 CHANNEL ACCESS SERVICE SDU

6.3.1 The abstract model of **Channel Access Service** is a queue at the sending-end **CA SAP**, the contents of which are transferred synchronously and periodically to the corresponding **CA SAP** on the ground. This model is illustrated in figure 6-2. Unlike other queued services at higher layers, CCSDS *Telemetry Channel Coding*, reference [2], does require a strict timing relationship between the sending **CA user** and the **Channel Access layer**. The **CA user** must provide **Transfer Frame**s to the **Channel Access Provider** at an average rate that will allow the **Channel Access Provider** to maintain a constant symbol rate to the Physical Access (r/f) layer below.

6.3.2 This model implies that

a) the **Transfer Frame**s sent by an on-board source are transferred in order;

b) the relationship to any other instance of **Channel Access Service** (i.e., another downlink channel on the same spacecraft) is not specified by CCSDS.

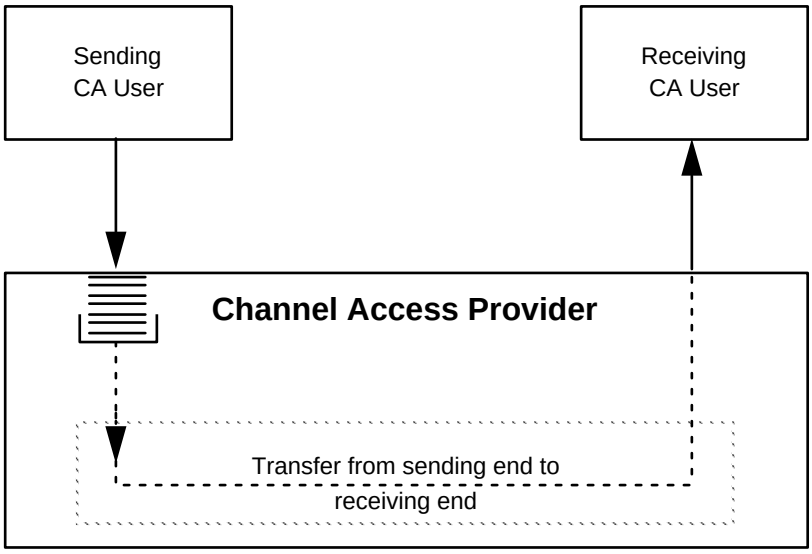


Figure 6-2: Abstract Model of Channel Access Service

6.4 PREREQUISITES FOR THE CHANNEL ACCESS SERVICE

6.4.1 The **Channel Access Service** requires **Physical Access Service** from the layer below. **Channel Access Service** also requires the specification of managed parameters, which serve to establish associations between space and ground protocol entities, specify address mappings, provide access authorization, and define operating limits.

6.4.2 Managed parameters for **Channel Access Service** include

a) fixed length of the **Transfer Frame**s on the Channel;

b) which, if any, of Reed-Solomon, randomization, differential modulation, or convolutional coding are used (plus related parameters, e.g., Reed-Solomon Interleaving depth—see reference [2]);

c) implementation-specific parameters for timing, latency, or flow control.

6.4.3 Neither this Recommendation nor reference [2] specifies methods, procedures, or formats for providing these managed parameters.

6.5 SERVICE PRIMITIVES OF THE CHANNEL ACCESS SERVICE

The service primitives associated with this service are

a) **CA.request**

The **CA.request** primitive is passed from the layer above at the sending end to the **Channel Access layer** to request that a **CA_SDU**, structured as a CCSDS **Transfer Frame**, be transferred.

b) **CA.indication**

The **CA.indication** is passed from the **Channel Access layer** at the receiving end to deliver a **CA_SDU**.

6.6 CHANNEL ACCESS SERVICE PARAMETERS

The parameters for the **Channel Access Service** primitives are described below.

a) **CA_SDU**	The **Channel Access SDU**. A **CA_SDU** is a fixed-length data unit, consisting of an integral number of octets.
b) **CA_Quality_Indication**	Indicator of data quality at the receiving end.
c) **CA_Sequence_Indication**	Indicator of continuity of synchronization at the receiving end.

6.7 DETAILED SPECIFICATION OF CHANNEL ACCESS SERVICE PRIMITIVES

6.7.1 General

This subsection describes in detail the primitives and parameters associated with the **Channel Access Service**. The parameters are specified in an abstract sense and specify the information to be made available to the user of the primitive. The way in which a specific implementation makes this information available is not constrained by this specification.

6.7.2 CA.request

a) Function:

The **CA.request** primitive is the service request primitive for the **Channel Access Service**.

b) Semantics:

The **CA.request** primitive shall provide parameters as follows:

CA.request (CA_SDU)

c) When Generated:

The **CA.request** primitive is passed to the **Channel Access layer** to request it to send the **CA_SDU** (**Transfer Frame**).

d) Effect on Receipt:

Receipt of the **CA.request** primitive causes the **Channel Access layer** to transfer the **CA_SDU**.

e) Additional Comments:

1) The **CA.request** primitive is used to transmit **Transfer Frames.**

2) The functions performed at the sending end when the **CA.request** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see sections 2 and 3 in *Telemetry Channel Coding*, reference [2].

Functions include: input fixed-length **Transfer Frame**; apply Reed-Solomon code (optional), randomization (optional), sync marker, conversion related to differential modulation (optional), and convolutional code (optional); provide a constant-rate stream of channel symbols to the **Physical Access layer** for modulation.

6.7.3 CA.indication

a) Function:

The **CA.indication** primitive is the service indication primitive for the **Channel Access Service**.

b) Semantics:

The **CA.indication** primitive shall provide parameters as follows:

CA.indication	**(CA_SDU,** **CA_Quality_Indicator,** **CA_Sequence_Indicator)**

c) When Generated:

The **CA.indication** primitive is passed from the **Space Transfer layer** to the **Channel Access Service** user to deliver a **CA_SDU**.

d) Effect on Receipt:

The effect of receipt of the **CA.indication** primitive by the user of the **Channel Access Service** is described in 5.2.7.3 e) 2).

e) Additional Comments:

1) The **CA.indication** primitive is used to deliver **Transfer Frame**s to the **CA user** process.

2) The functions performed at the receiving end before the **CA.indication** primitive is sent are summarized below. This functional overview is not part of this Recommendation; see sections 2 and 3 in *Telemetry Channel Coding*, reference [2].

Functions include: input constant-rate stream of channel symbols; apply frame sync, decoding/de-randomizing (optional), conversion related to differential modulation (optional); deliver **Transfer Frame**, with **Channel Access** sequence indication and **Channel Access** quality indication (optional).

ANNEX A

ACRONYMS

(This annex **is not** part of the Recommendation.)

APID	Application Process Identifier
BRM	Basic Reference Model
CC	Channel Coding
CCSDS	Consultative Committee for Space Data Systems
CLCW	Command Link Control Word
FECF	Frame Error Control Field
FHP	First Header Pointer
FSH	Frame Secondary Header
GSCID	Global Spacecraft Identifier
GVCID	GSCID concatenated with the VCID
MC	Master Channel
MC_FSH	MC Frame Secondary Header
MC_OCF	MC Operational Control Field
NASA	National Aeronautics and Space Administration
OCF	Operational Control Field
OSI	Open System Interconnection
PDD	Privately Defined Data
PDU	protocol-data-unit
PT	Packet Telemetry
PTS	Packet Telemetry Service
PVN	Packet Version Number
SAP	service-access-point
SDU	service-data-unit
SP	Source Packet
TF	Transfer Frame
VC	Virtual Channel
VCID	Virtual Channel Identifier

ANNEX B

INFORMATIVE REFERENCES

(This annex **is not** part of the Recommendation.)

[B1] *Procedures Manual for the Consultative Committee for Space Data Systems*. CCSDS A00.0-Y-7. Yellow Book. Issue 7. Washington, D.C.: CCSDS, November 1996.

[B2] *Telemetry Summary of Concept and Rationale*. Report Concerning Space Data Systems Standards, CCSDS 100.0-G-1. Green Book. Issue 1. Washington, D.C.: CCSDS, December 1987.

[B3] *Information Technology—Open Systems Interconnection—Basic Reference Model: The Basic Model*. International Standard, ISO/IEC 7498-1. 2nd ed. Geneva: ISO, 1994.

[B4] *Information Processing Systems—Open Systems Interconnection Reference—Basic Reference Model—Part 3: Naming and Addressing*. International Standard, ISO 7498-3. 1st ed. Geneva: ISO, 1989.

[B5] *Information Technology—Open Systems Interconnection—Basic Reference Model—Conventions for the Definition of OSI Services*. Draft International Standard, ISO/IEC DIS 10731. Geneva: ISO, 1991.

ANNEX C

A BRIEF TUTORIAL ON OSI SERVICE TERMINOLOGY

(This annex **is not** part of the Recommendation.)

C1 OSI SERVICE MODELING CONVENTIONS

The OSI Basic Reference Model (BRM) (reference [B3]) introduces the OSI Environment (OSIE), which is the set of concepts, elements, services, and protocols that allow communication among open systems. An open system is the representation of those aspects of a real open system that are pertinent to interconnection with other (real) open systems. A real open system is a real system that complies with OSI standards, where a real system is "a set of one or more computers, the associated software, peripherals, terminals, human operators, physical processes, information transfer means, etc., that forms an autonomous whole capable of performing information processing and/or information transfer."

Implied but not explicitly stated in the OSI documentation is the fact that real (open) systems exist to support applications. The aspect of the application that is of interest with respect to OSI is the **application process**, which is the "element within a **real open system** which performs the information processing for a particular application."

Although OSI facilitates the interconnection of open systems, and thus the interactions of application processes, OSI is not concerned with the specification of those application processes. According to reference [B3]:

- "OSI is concerned with the exchange of information between open systems (and not the internal functions of each individual real open system)." (4.2.7)

- "OSI is concerned only with the interconnection of systems. All other aspects of systems which are not related to interconnection are outside the scope of OSI." (4.2.9)

- "OSI is concerned not only with the transfer of information between systems, i.e., transmission, but also with their capability to interwork to achieve a common (distributed) task. In other words, OSI is concerned with the interconnection aspects of cooperation between systems, which is implied by the expression 'open systems interconnection.'" (4.2.10)

The OSI documents address a range of topics regarding OSI services. For the purposes of this Recommendation, these topics fall into three categories: the OSI service definition conventions, the OSI BRM definitions, and the OSI management framework.

C2 OSI SERVICE DEFINITION CONVENTIONS

Since OSI is concerned with providing interconnections between (or among) application processes, it follows that from the OSI perspective, *service* is defined in terms of providing the connection (and not, for example, in terms of what one application process on one open system does to support the application process on another open system). The formal definitions of **OSI-service** and related terms are not found in the BRM proper but rather in the OSI service definitions convention standard [B4]. Figure C-1, which is a copy of figure 1 in [B4], illustrates the OSI service model. As shown, three **OSI-service-users** exchange **OSI-service-primitives** (**submit** primitives and **deliver** primitives) with the **OSI-service-provider**. A **submit** primitive is an **OSI-service-primitive** initiated by an **OSI-service-user**. A **deliver** primitive is an **OSI-service-primitive** initiated by an **OSI-service-provider**. The shared behavior of an **OSI-service-user** and an **OSI-service-provider** in terms of their interactions at the service boundary is called the **OSI-local-view**. When an **OSI-service** is defined such that all of its **OSI-local-views** are the same (i.e., there is only one type of **OSI-local-view** for the service), the **OSI-service** is said to be a **symmetrical-service**. When an **OSI-service** is defined such that all of its **OSI-local-views** are the not the same (i.e., there are several types of **OSI-local-view** for the service), the **OSI-service** is said to be an **asymmetrical-service**.

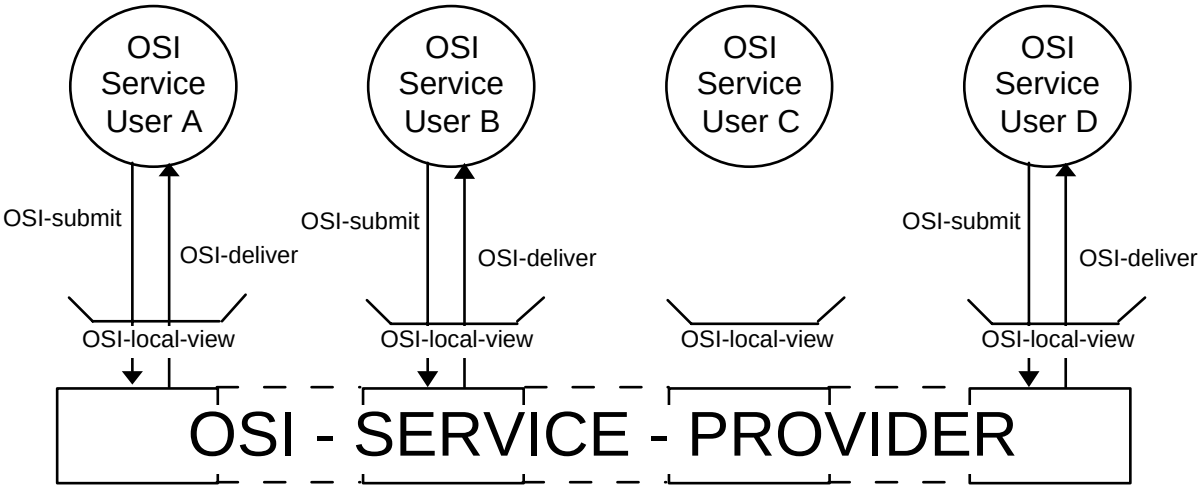


Figure C-1: OSI-Service Model

At an **OSI-local-view**, related **submit** and **deliver** primitives form **OSI-service-procedures**. An **OSI-service-procedure** is "either a **submit** primitive together with the locally resulting **deliver** primitive or primitives, if any, or a **deliver** primitive together with the locally resulting **submit** primitive or primitives, if any, seen at an **OSI-local-view**."

The OSI service model provides limited terminology for describing the roles of the two (or more) users of an OSI service. These roles are **requestor** and **acceptor**. In the context of a particular instance of **OSI-service-procedure**, a **requestor** is "an **OSI-service-user** that

issues a **submit** primitive and as a result may receive one or more **deliver** primitives." In the context of a particular instance of **OSI-service-procedure**, an **acceptor** is "an **OSI-service-user** that receives a **deliver** primitive and as a result may issue one or more **submit** primitives." Names are defined for the **submit** and **deliver** primitives used by the **requestor** and **acceptor**: **request**, **indication**, **response**, and **confirm**.

- A **request** is a **submit** primitive issued by a **requestor**.

- An **indication** is a **deliver** primitive received by an **acceptor**. The **indication** is related (in a way that is specific to the **OSI-service**) to the **request** issued by the **requestor**.

- A **response** is a **submit** primitive issued by an **acceptor** as a result of the **indication** received.

- A **confirm** is a **deliver** primitive received by the **requestor**. The **confirm** is related (in a way that is specific to the **OSI-service**) to the **response** issued by the **acceptor**.

For connectionless-mode OSI-services, only the request and indication have meaning, since a connectionless-mode OSI service does not maintain the state (that is, track the relationship) between data flowing from **requestor**-to-**acceptor** and data flowing from **acceptor**-to-**requestor**. Figure C-2, derived from figure 3 of reference [B4], illustrates the user roles and primitives present for a connection-mode **OSI-service**.

Figure C-2: Example of a Peer-to-Peer Connection-Mode Service

C3 OSI BRM DEFINITIONS

The OSI service definition conventions provides a general frame of reference for describing the roles and relationships between a user of an interconnection service and a provider of

such a service. The OSI-BRM builds upon these ISO service definition conventions to address modeling of interconnections between open systems. In an open systems environment the **OSI-service-provider** is realized through the interaction of service entities residing in the open systems. Furthermore, the OSI-BRM employs the concept of a layered architecture. In this architecture, each open system is viewed as being logically composed of an ordered set of **(N)-subsystems**, where *N* denotes the number of the layer (the **(N)-layer**) in which the logical subsystem exists. Figure C-3 illustrates the realization of an OSI-service-provider in the layered open system environment, using key terminology.

As shown in figure C-3, one or more **(N)-entities** exist within each **(N)-subsystem**, where the **(N)-entity** is an active element that embodies a set of capabilities defined for the **(N)-layer**. **Peer-(N)-entities** within each **(N)-layer** interact according to **(N)-protocols** to form the **(N)-service-provider**, which is the **OSI-service-provider** at the **(N)-layer**. The **peer-(N)-entities** interact via the exchange of **(N)-protocol-data-units**. The primitives exchanged between the **(N)-service-user** and its local **(N)-entity** contain **(N)-user-data** in the form of **(N)-service-data-units**. In addition to the **(N)-user-data**, the service primitives contain information needed by the **(N)-entity** to properly execute the service.

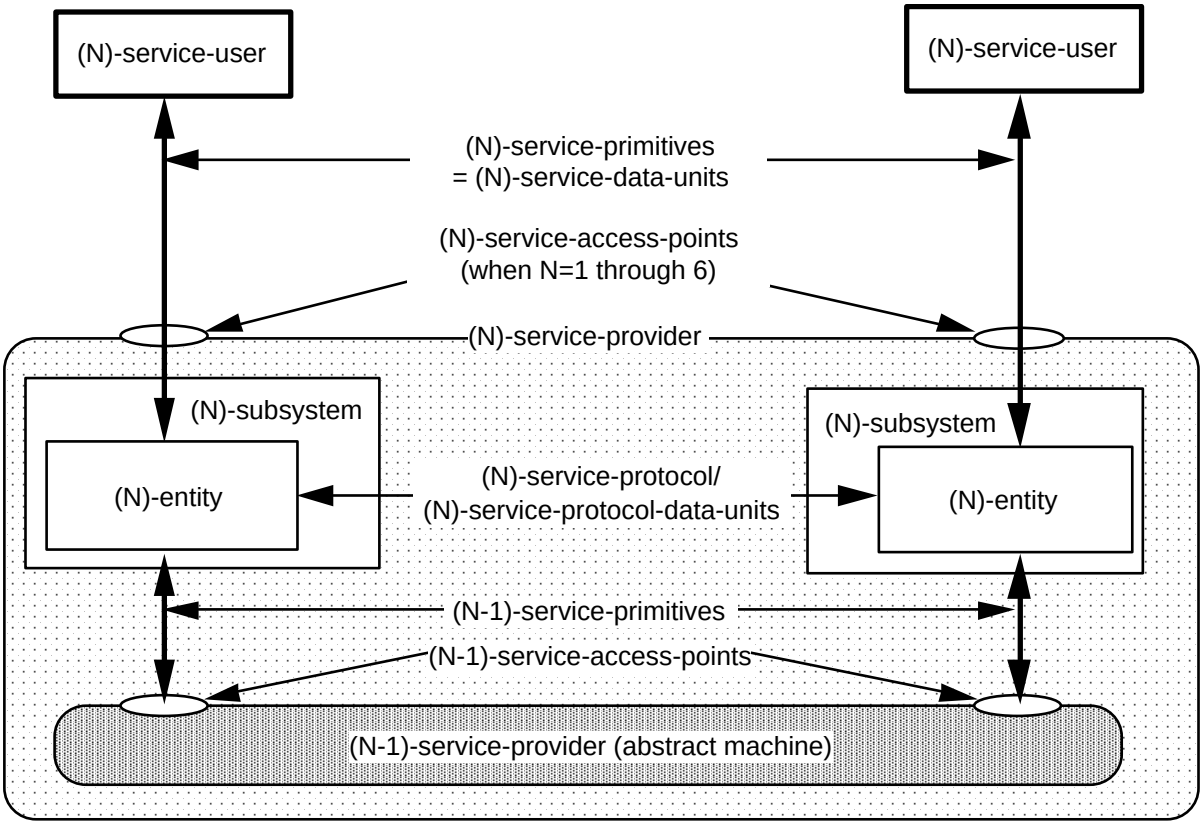


Figure C-3: OSI-Service Components on Open Systems

In the OSI-BRM layered architecture, the exchange of **(N)-protocol-data-units** between the **(N)-entities** is carried out by employing the **OSI-service** of the layer below. With respect to an **(N-1)-service-provider**, the **(N)-entity** is also the **(N-1)-service-user**.

For the OSI physical through presentation layer services (i.e., N = 1 through 6), the **(N)-entity** provides the **(N)-service** to the **(N)-service-user** at the **(N)-service-access-point**. The concept of service access point does not apply to services provided by the application layer.

In addition to defining how layered services are realized on distributed open systems, the BRM defines a specific set of layers—the infamous seven layers—and defines the functionality of each of those layers.