BS ISO 20217:2015

BSI Standards Publication

Space data and information transfer systems — Spacecraft onboard interface services — File and packet store services

bsi.

...making excellence a habit.™

Document Sharing Hub

National foreword

This British Standard is the UK implementation of ISO 20217:2015.

The UK participation in its preparation was entrusted to Technical Committee ACE/68, Space systems and operations.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2015.
Published by BSI Standards Limited 2015

ISBN 978 0 580 87916 6

ICS 49.140

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 December 2015.

Amendments/corrigenda issued since publication

Date	Text affected

Documents
Document Sharing Hub

Full document details: BS ISO 20217:2015

INTERNATIONAL STANDARD

ISO 20217

First edition
2015-08-15

Space data and information transfer systems — Spacecraft onboard interface services — File and packet store services

Systèmes de transfert des informations et données spatiales — Services d'interfaces à bord des véhicules spatiaux — Services de mémoire fichiers et mémoire paquets

Reference number
ISO 20217:2015(E)

Documents
Document Sharing Hub

© ISO 2015

Full document details: BS ISO 20217:2015

COPYRIGHT PROTECTED DOCUMENT

© ISO 2015, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub

Full document details: BS ISO 20217:2015

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 20217 was prepared by the Consultative Committee for Space Data Systems (CCSDS) (as CCSDS 873.0-M-1, September 2012) and was adopted (without modifications except those stated in clause 2 of this International Standard) by Technical Committee ISO/TC 20, *Aircraft and space vehicles*, Subcommittee SC 13, *Space data and information transfer systems*.

Documents
Document Sharing Hub

© ISO 2015 – All rights reserved

Full document details: BS ISO 20217:2015

Documents
Document Sharing Hub

The Consultative Committee for Space Data Systems

Recommendation for Space Data System Practices

SPACECRAFT ONBOARD INTERFACE SERVICES—FILE AND PACKET STORE SERVICES

RECOMMENDED PRACTICE

CCSDS 873.0-M-1

MAGENTA BOOK

September 2012

Documents
Document Sharing Hub

Full document details: BS ISO 20217:2015

AUTHORITY

Issue:	Recommended Practice, Issue 1
Date:	September 2012
Location:	Washington, DC, USA

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and represents the consensus technical agreement of the participating CCSDS Member Agencies. The procedure for review and authorization of CCSDS documents is detailed in *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-3), and the record of Agency participation in the authorization of this document can be obtained from the CCSDS Secretariat at the address below.

This document is published and maintained by:

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

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

STATEMENT OF INTENT

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

CCSDS Recommendations take two forms: **Recommended Standards** that are prescriptive and are the formal vehicles by which CCSDS Agencies create the standards that specify how elements of their space mission support infrastructure shall operate and interoperate with others; and **Recommended Practices** that are more descriptive in nature and are intended to provide general guidance about how to approach a particular problem associated with space mission support. This **Recommended Practice** is issued by, and represents the consensus of, the CCSDS members. Endorsement of this **Recommended Practice** is entirely voluntary and does not imply a commitment by any Agency or organization to implement its recommendations in a prescriptive sense.

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

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

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

FOREWORD

This document is a technical Recommended Practice for use in developing flight and ground systems for space missions and has been prepared by the Consultative Committee for Space Data Systems (CCSDS). The File and Packet Store Services described herein is intended for missions that are cross-supported between Agencies of the CCSDS, in the framework of the Spacecraft Onboard Interface Services (SOIS) CCSDS area.

This Recommended Practice specifies a set of related services to be used by space missions to access and manage files and packets within a spacecraft subnetwork. The SOIS File and Packet Store Services provide a common service interface regardless of the particular type of data link or protocol being used for communication.

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Recommended Practice is therefore subject to CCSDS document management and change control procedures, which are defined in the *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-3). Current versions of CCSDS documents are maintained at the CCSDS Web site:

http://www.ccsds.org/

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

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

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

Member Agencies

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

Observer Agencies

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

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

DOCUMENT CONTROL

Document	Title	Date	Status
CCSDS 873.0-M-1	Spacecraft Onboard Interface Services—File and Packet Store Services, Recommended Practice, Issue 1	September 2012	Current issue

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

CONTENTS

Section Page

1 INTRODUCTION 1-1

1.1 PURPOSE AND SCOPE OF THIS DOCUMENT 1-1
1.2 APPLICABILITY 1-1
1.3 RATIONALE 1-1
1.4 DOCUMENT STRUCTURE 1-1
1.5 CONVENTIONS AND DEFINITIONS 1-2
1.6 HOW THIS DOCUMENT FITS INTO THE SOIS DOCUMENTATION TREE . 1-4
1.7 CONVENTIONS 1-4
1.8 REFERENCES 1-4

2 SERVICE CONCEPT 2-1

2.1 OVERVIEW 2-1
2.2 FILE STORE OVERVIEW 2-2
2.3 PACKET STORE OVERVIEW 2-4
2.4 PURPOSE AND OPERATION OF THE FILE AND PACKET STORE SERVICES 2-8
2.5 EXAMPLE DEPLOYMENT 2-10

3 SERVICE DEFINITION 3-1

3.1 PROVIDED SERVICE 3-1
3.2 SERVICES EXPECTED FROM THE UNDERLYING LAYERS 3-4
3.3 SERVICE PARAMETERS 3-9
3.4 SERVICE INTERFACE 3-15

4 MANAGEMENT INFORMATION BASE 4-1

4.1 DISCUSSION 4-1
4.2 SPECIFICATIONS 4-1
4.3 MIB GUIDANCE 4-1
4.4 MAXIMUM FILE SIZE 4-1
4.5 MAXIMUM PACKET STORE SIZE 4-1

ANNEX A FILE AND PACKET STORE SERVICES PROTOCOL IMPLEMENTATION CONFORMANCE STATEMENT PROFORMA (NORMATIVE) A-1
ANNEX B SECURITY CONSIDERATIONS (INFORMATIVE) B-1
ANNEX C ACRONYMS (INFORMATIVE) C-1
ANNEX D INFORMATIVE REFERENCES (INFORMATIVE) D-1
ANNEX E TYPICAL API—POSIX (INFORMATIVE) E-1

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

CONTENTS (continued)

Figure Page

2-1 File and Packet Store Services Context 2-1
2-2 Random-Access Packet Store Logical Structure 2-6
2-3 Dumping Packets from the Packet Store System to the Spacecraft Telemetry Chain 2-7
2-4 Example Deployment of FPSS and File and Packet Store Systems 2-9
2-5 Multiple Distributed Mass Memories in a File Store 2-10
2-6 Example Interfacing from SSMM-based Packet Store System to Telemetry System 2-11
2-7 Example Interfacing from OBC-based Packet Store System to Telemetry System 2-11

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

1 INTRODUCTION

1.1 PURPOSE AND SCOPE OF THIS DOCUMENT

This document defines the Spacecraft Onboard Interface Services (SOIS) File and Packet Store Services (FPSS). The definition encompasses specification of the service interface exposed to onboard software (user applications and libraries) as well as the conceptual mapping of the FPSS primitives to the protocols implementing such services.

The SOIS File and Packet Store Services are for use by onboard software to:

- Access, and manage files residing in a file store. The files residing in the file store could contain any type of data, including for example telemetry, commands and command sequences, software updates, imagery, and other science observations.
- Access and manage packets residing in a packet store. It should be noted that the packet store may or may not be aware of, and its actions may or may not be informed by, the contents of a packet.

To achieve this, the FPSS comprise the following services:

- File Access Service (FAS);
- File Management Service (FMS);
- Packet Store Access Service (PSAS);
- Packet Store Management Service (PSMS).

It should be noted that the SOIS File and Packet Store Services do NOT define the file and packet stores themselves, but only their minimum provided service. The initialization and configuration of the file and packet stores is out of scope for this document.

1.2 APPLICABILITY

This document applies to any mission or equipment claiming to provide CCSDS SOIS-compatible File and Packet Store Services.

1.3 RATIONALE

SOIS provides service interface specifications in order to promote commonality of functionality amongst systems implementing well-defined services. These interfaces do not dictate implementation of interfaces or protocols supporting the services.

1.4 DOCUMENT STRUCTURE

This document has four major sections:

- section 1, this section, containing administrative information, definitions, and references;

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

- section 2, containing the general concepts and assumptions;
- section 3, containing the File and Packet Store Services, in terms of the services provided, services expected from underlying layers, and the service interface;
- section 4, containing the Management Information Base (MIB) for this service.

In addition, one normative and four informative annexes are provided:

- annex A, comprising a Service Conformance Statement Proforma;
- annex B, containing a discussion security considerations with respect to the specifications of this document;
- annex C, containing a list of acronyms;
- annex D, containing a list of informative references;
- annex E, describing aspects of the POSIX API file functions which might be useful to implementers of the File Access and Management Services.

1.5 CONVENTIONS AND DEFINITIONS

1.5.1 BIT NUMBERING CONVENTION AND NOMENCLATURE

In accordance with modern data communications practice, spacecraft data fields are often grouped into eight-bit 'words' widely known as bytes. Throughout this Recommended Practice, such an eight-bit word is called an 'octet'. The numbering for octets within a data structure starts with zero.

By CCSDS convention, any 'spare' bits shall be permanently set to '0'.

1.5.2 DEFINITIONS

1.5.2.1 Definitions from the Open Systems Interconnection (OSI) Reference Model

The document is defined using the style established by the Open Systems Interconnection (OSI) Basic Reference Model (reference [D1]). This model provides a common framework for the development of standards in the field of systems interconnection.

The following terms used in this Recommended Practice are adapted from definitions given in (reference [D1]):

layer: A subdivision of the architecture, constituted by subsystems of the same rank.

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

service: A capability of a layer, and the layers beneath it (service providers), provided to the service users at the boundary between the service providers and the service users.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

(N)-service-access-point, (N)-SAP: The point at which (N)-services are provided by an (N)-entity to an (N+l)-entity. Within the spacecraft, a SOIS User Service Access Point. As a minimum it locates a data system and an application within that data system.

1.5.2.2 Terms defined in this Recommended Practice

For the purposes of this Recommended Practice, the following definitions also apply:

application: Any component of the onboard software that makes use of the File and Packet Store Services. This includes flight software applications and higher-layer services.

file: An ordered collection of arbitrary information, or resource for storing information, residing in some kind of durable storage, and with an associated name.

file system: A method of storing and organizing files and the data they contain to make it easy to find and access them. File systems may use a local data storage device or provide access to data on a remote server over a network. Optionally, there may be a hierarchy of files, i.e., a folder or directory tree.

file store: A file system and its storage media. A file store comprises:

- one or more *mass memories* in which files reside;
- an associated *file system* providing services for managing the files stored in the mass memories. This document does not intend to define a file system; it is therefore assumed that a file system interfacing to one or more mass memories is already present.

NOTE – No assumption is made about persistence of files in the file store or any file replication strategies.

packet: At a minimum level of complexity, a delimited data structure. Examples of such packets are CCSDS Space Packets (reference [D4]), CCSDS Encapsulation Packets (reference [D5]), IPv4 (reference [D6]) and IPv6 (reference [D7]) packets. Packets are time-stamped with the time the packet was stored in the packet store. For CCSDS packets, primary header fields are accessible to the PSAS. These include packet identifiers and protocol identifiers.

packet store: A collection of packets and its storage media. A packet store comprises:

- one or more *mass memories* in which packets reside;
- an associated *packet storage system* providing services for managing the packets stored in the mass memories.

NOTES

1 No assumption is made about persistence of packets in the packet store or any replication strategies.

2 No assumption is made about where the packet store is located (remotely or locally).

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

octet: An eight-bit word.

value: A formatted atomic unit of data that is stored in a file or a packet.

1.6 HOW THIS DOCUMENT FITS INTO THE SOIS DOCUMENTATION TREE

This document conforms to the principles set out in the Spacecraft Onboard Interface Services Green Book (reference [D2]) and should not be applied without first consulting this reference.

1.7 CONVENTIONS

1.7.1 NOMENCLATURE

The following conventions apply for the normative specifications in this Recommended Practice:

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

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

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

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

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

1.7.2 INFORMATIVE TEXT

In the normative sections of this document (sections 3-4 and annex A), informative text is set off from the normative specifications either in notes or under one of the following subsection headings:

– Overview;

– Background;

– Rationale;

– Discussion.

1.8 REFERENCES

This document contains no normative references. Informative references are contained in annex D.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

2 SERVICE CONCEPT

2.1 OVERVIEW

The SOIS File and Packet Store Services (FPSS) are defined within the context of the overall SOIS architecture (see reference [D2]) as one of the services of the Application Support Layer, as illustrated in figure 2-1.

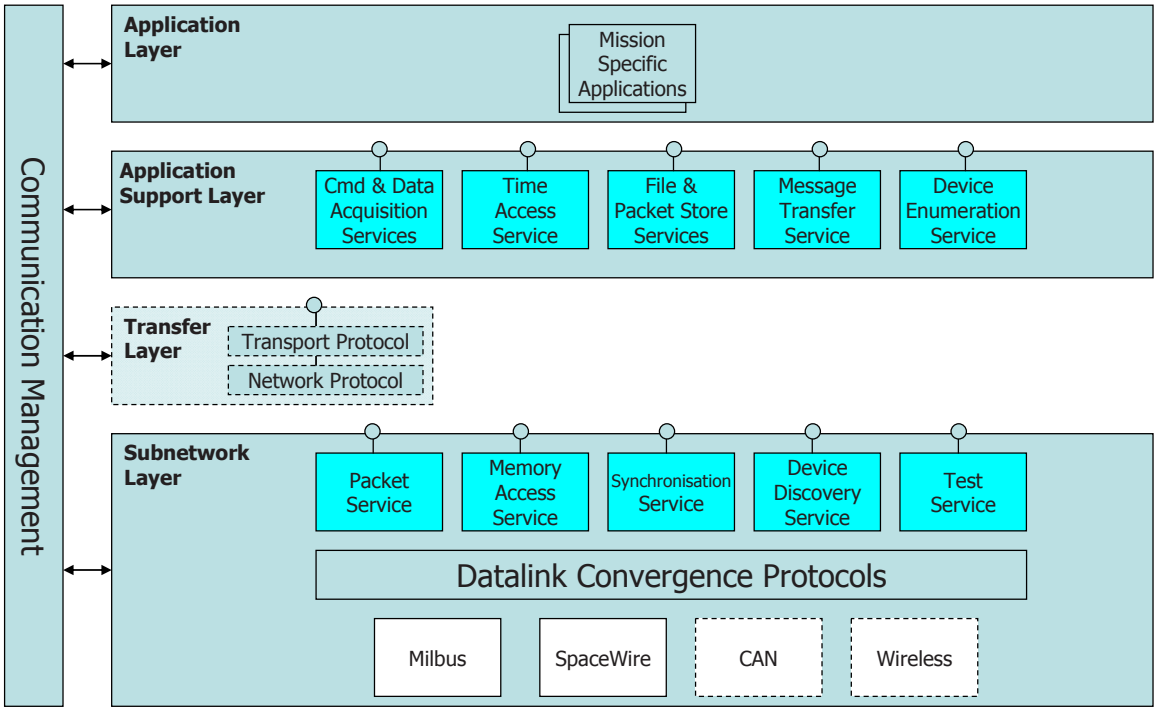

Figure 2-1: File and Packet Store Services Context

The SOIS File and Packet Store Services provide a standard interface to allow onboard software to request:

- access to files resident in a file store;
- modifications to files in a file store;
- management of files within a file store (e.g., create, delete, rename);
- storing, retrieving, and deleting packets in a packet store;
- management of packet stores.

The basic concept underlying the service is that the onboard software should be able to access files in a file store or packets in a packet store independently of the precise physical location or address of the store within the spacecraft's communication architecture, and without requiring detailed knowledge of the mechanism used to access or implement the store. A standard interface makes it easier to develop onboard software, enables configuration changes in the spacecraft design to be easily tolerated, and increases the re-use potential of the software.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

2.2 FILE STORE OVERVIEW

2.2.1 TYPES OF FILE STORES

A file store is considered to be a file system and the associated storage medium.

At the highest level a file system is a way to organize, store, retrieve, and manage information on a permanent storage medium. File stores manage permanent storage and form an integral part most onboard spacecraft data handling systems.

Different types of file systems (and file stores) may be considered:

a) **Flat file systems**. Designed to not have subdirectories. Everything is stored in the root level of the storage media.

b) **Hierarchical file systems**. Designed to organize information in directories in a tree-like structure.

Either of these may be used to support the FPSS services and no restriction is imposed on the type of file store. This approach allows complete independence from the technology used to implement the file store. The way in which this mapping is performed is implementation-specific.

The FPSS allow for multiple client processes on multiple machines for both accessing and updating files. Hence updates to the file from one client should not interfere with access and updates from other clients. It is assumed that *concurrency control* or *locking* is handled by the file store itself. In support of this, file locking functionality is provided. An application (the 'lock owner') may lock a file to various degrees of access (exclusive read-only, read-only by all, exclusive access) from other user entities. Any user entity may unlock the file.

NOTE – It is implementation-specific if unlocking of files is restricted to, for example, a 'super-user' application.

2.2.2 CURRENT DIRECTORY

The **current directory** within a file store for each application is maintained by the file system. A number of operations are performed relative to an application's current directory, e.g., creating a file. By default, upon initialization, the application's current directory is at the root level of the storage media for flat file systems and at the root level of the tree for hierarchical file systems. For hierarchical file systems, each application's current directory can be changed by request.

2.2.3 TYPES OF FILES

A file is where a user stores information. Files are logically organized into one-dimensional arrays of octets. The format of a file is defined by its content since a file is solely a container

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

for data. The size of data stored is arbitrary and may range from only a few octets to the entire capacity of the entire mass memory in which files reside. Files are not limited in size (subject to the limitations of the mass memory in which the files reside). A file store should be able to hold a potentially large number of files, where 'large' ranges from tens of thousands to millions.

Different types of files may be considered:

a) **Regular**. Regular files are the most common files and are used to contain data, for example, information stored in ASCII format text and readable by the user, or information in computer-readable formats (e.g., software images).

b) **Directory**. A directory is a file system structure that contains information that the system needs to access all types of files, but directories do not contain actual file data. Directories permit convenient management of the file system structure. Directories are created and controlled by a separate set of commands.

It is assumed that the file store has regular files. Directory files are optional for SOIS purposes.

2.2.4 BASIC FILE OPERATIONS

There is a set of basic operations that may be performed upon a file in a file system: open, read, write, append, seek, and close.

The **current position** within an open file for each application is maintained by the file system. By default, upon opening a file, the application's current position in the file is initialized to the start of the file. Reading data from and writing data to files is performed at the application's current position, and the application's current position is incremented by the length of data read or written.

Writing to a file overwrites the original data with new data, unless the application's current position is at the end of the file (i.e., beyond the last data in the file), in which case the new data is appended to the end of the file. Reading when the application's current position is at the end of the file returns no data.

The application's current position can be moved to a specified offset from the start of the file using a seek operation, thereby providing random access to the file. This may, for example, be used to overwrite part of a file's contents or to read from a part of a file.

Finally, upon opening a file, the application may indicate that the file is being opened to be appended to, in which case the application's current position in the file is initialized to the end of the file.

Although this Recommended Practice does not exclude a file's being open for simultaneous writing by multiple user entities, it is strongly advised that only a single application at any one time should have write access.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

2.3 PACKET STORE OVERVIEW

2.3.1 GENERAL

Any storage technology that is suitable for spacecraft missions may be used to implement a packet store. This approach allows complete independence from the technology used to implement the packet store. The way in which this mapping is performed is implementation-specific.

The FPSS should provide for multiple client processes on multiple machines accessing the same packet store. Hence access from one client should not interfere with accesses from other clients. It is assumed that concurrency control or locking is handled by the packet store itself.

The packet store may provide a separate interface to the spacecraft telemetry system that permits packets to be 'dumped' (i.e., sent directly) to the ground using the spacecraft telemetry system (see 2.3.4).

2.3.2 TYPES OF STORED PACKETS

The packet store may or may not be aware of the contents of a packet. A stored packet is, at a minimum level of complexity, just a delimited data structure.

Examples of such packets are the CCSDS Space Packet (reference [D4]), the CCSDS Encapsulation Packet (reference [D5]), IPv4 (reference [D6]) packets, and IPv6 (reference [D7]) packets. Packet stores are tagged as containing either CCSDS packets or non-CCSDS packets as searches on structural elements of a packet can only be performed on the primary header of CCSDS Space and CCSDS Encapsulation Packets (see 2.3.3.2.2).

Any storage technology that is suitable for spacecraft missions may be used to implement a packet store. This approach allows complete independence from the technology used to implement the packet store. The way in which this mapping is performed is implementation-specific.

The FPSS should provide for multiple client processes on multiple machines accessing the same packet store. Hence access from one client should not interfere with accesses from other clients. It is assumed that *concurrency control* or *locking* is handled by the packet store itself.

2.3.3 TYPES OF PACKET STORES

Packet stores onboard a spacecraft can be organized in different ways, with the most common being:

a) **First-in, First-Out** (FIFO):

   1) bounded;

   2) circular;

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

b) **random-access**.

Other packet store types exist, but this specification covers only the above mentioned types. The following subsections provide more detail on these common types.

2.3.3.1 FIFO Packet Store

The FIFO Packet Store is used to store packets in a FIFO queue; i.e., packets are written to the tail of the queue and read from the head of the queue.

As the packet store is a FIFO, only the packet sizes have to be known, not their structure.

Writing to and reading from the packet store is always performed in sequential order, writing new packets to the tail, reading the oldest unread packets from the head, and freeing the oldest packets from the head of the queue.

Although this Recommended Practice does not exclude multiple user entities' reading, dumping, and freeing from the packet store, it is strongly advised that only a single application should read, dump, and free from the packet store. FIFO Packet Stores can be of two forms:

– **Bounded**. A bounded FIFO Packet Store has a defined maximum number of packets that can be stored in its queue. When the queue is full, no more packets can be added to the tail of the queue: packets in the Packet Store are NEVER removed/discarded by the service in order to store new packets; instead, the user must explicitly free them. When freeing from the head of the queue, the packet is removed and a space freed up at the tail for a new packet.

– **Circular**. A circular FIFO Packet Store also has a defined maximum number of packets that can be stored in its queue. However, unlike the bounded FIFO Packet Store, when attempting to write to the tail of a full queue, the packet at the head of the queue, i.e., the oldest, is deleted. This kind of packet store can never run out of space to store new packets but results in the possibility of packets being discarded by the service.

2.3.3.2 Random-Access Packet Store

In addition to the FIFO queue of a FIFO packet store, a random-access Packet Store allows selective read and delete access to stored packets. This may for example be used for payload-specific telemetry.

The logical structure of a random-access packet store is illustrated in figure 2-2. The packets in the packet store are stored in a time-stamped (see 2.3.3.2.1) FIFO queue, with a write pointer at the tail of the queue (i.e., the newest packet), a read pointer in the queue pointing to the next packet to be read, and a free pointer at the head of the queue (i.e., the oldest packet). When a packet is written to the packet store, the packet is added to the tail of the queue and the write pointer modified to point to the packet. When a packet is read from the packet store, the packet pointed to by the read pointer is returned to the user and the read

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

pointer is moved to the next packet in the queue. When a packet is freed from the packet store, any storage associated with the packet pointed to by the free pointer is released and the free pointer is moved to the next packet in time order.

A selection of packets can be made, based on selection criteria (see 2.3.3.2.2). This is also held as a time-stamped FIFO queue (the 'selection queue') of references to the packets in the main FIFO queue. A selective read pointer points to the next packet to be read from the selection. A selective free pointer points to the head of the selection queue. When a packet is selectively read from the packet store, the packet pointed to by selective read pointer is returned and the selective read pointer moved to the next reference in the selective queue. When a packet is selectively freed from the packet store, any storage associated with the packet referenced by the selective free pointer is released and the selective free pointer is moved to the next reference in the selection queue (potentially updating any pointers associated with the packet store, e.g., where the freed packet was also the oldest packet in the packet store).

A request to free packets from the packet store may free packets referenced in the selection queue. Therefore, the selection will be marked invalid and any resources associated with the selection queue released.

Although this Recommended Practice does not exclude multiple user entities' using packet selections, it is strongly advised that only a single application at any one time should use packet selections.

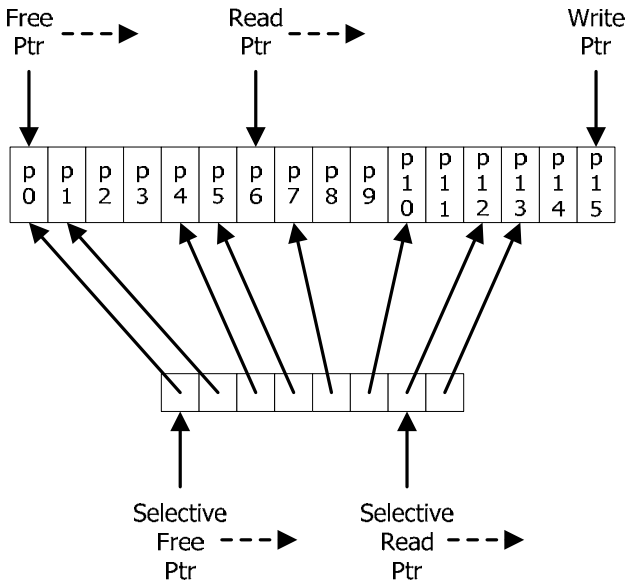

Figure 2-2: Random-Access Packet Store Logical Structure

NOTE – Having selective delete raises the problem of the **packet store fragmentation**. The implementation of the service should therefore take into account a cleaning policy to compact the live data in order to avoid any performance degradation due to progressive fragmentation.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

2.3.3.2.1 Time Stamping of Packets

The time stamp associated with a packet will be the time when the packet was written to the packet store. This is only applied to either FIFO or random access packet stores.

NOTE – A time stamp may also be present in a CCSDS packet's secondary header. Packet Store implementations may have knowledge of this; however, this is not covered by this CCSDS Recommended Practice.

2.3.3.2.2 Packet Selection Criteria

The applicable selection criteria (only applied to random access packet stores) are as follows:

a) packet store '*write time*';

b) one or more structural elements within the primary header of the CCSDS packet (valid only for packet stores containing CCSDS Packets).

NOTES

1 The selection criteria used for a single field is dependent upon the particular field and is implementation-dependent, e.g., an exact match to the APID field.

2 A Packet Store implementation may have knowledge of the contents of a secondary header of the CCSDS packet; however, this is not covered by this CCSDS Recommended Practice.

2.3.4 DUMPING OF TELEMETRY PACKETS

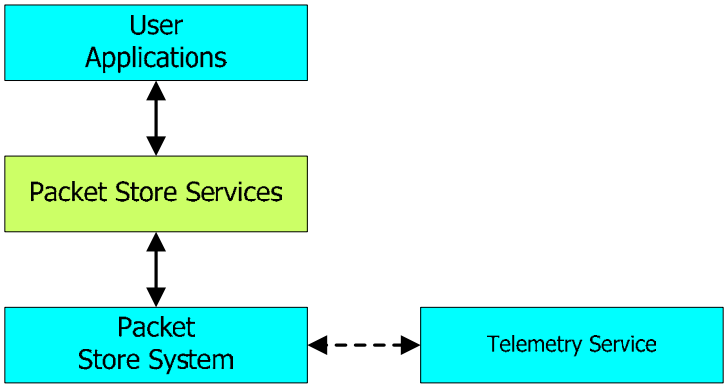

Figure 2-3: Dumping Packets from the Packet Store System to the Spacecraft Telemetry Chain

A number of packet store systems support the direct dumping of packets to the spacecraft telemetry chain, this being more efficient than a packet store service user reading packets and then sending them to the spacecraft telemetry system.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

The packet store services provide interfaces for dumping and selectively dumping packets directly from the packet store system to the spacecraft telemetry system.

NOTE – The implementation of the interface from the packet store system to the spacecraft telemetry chain is not specified here. It may or may not make use of other SOIS services.

2.4 PURPOSE AND OPERATION OF THE FILE AND PACKET STORE SERVICES

Each service provides a consistent, standard interface to onboard software; the interfaces are described by sets of primitives and related parameters.

From the user's perspective, use of the FPSS will result in onboard software that is more portable, easier to develop, and more tolerant of changes in the spacecraft hardware configuration.

From the spacecraft platform implementer's perspective, use of the FPSS will make it easier to control access and management of shared hardware resources (i.e., mass memories).

The FPSS are operated using service requests and service indications passed between the service user and the service provider.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

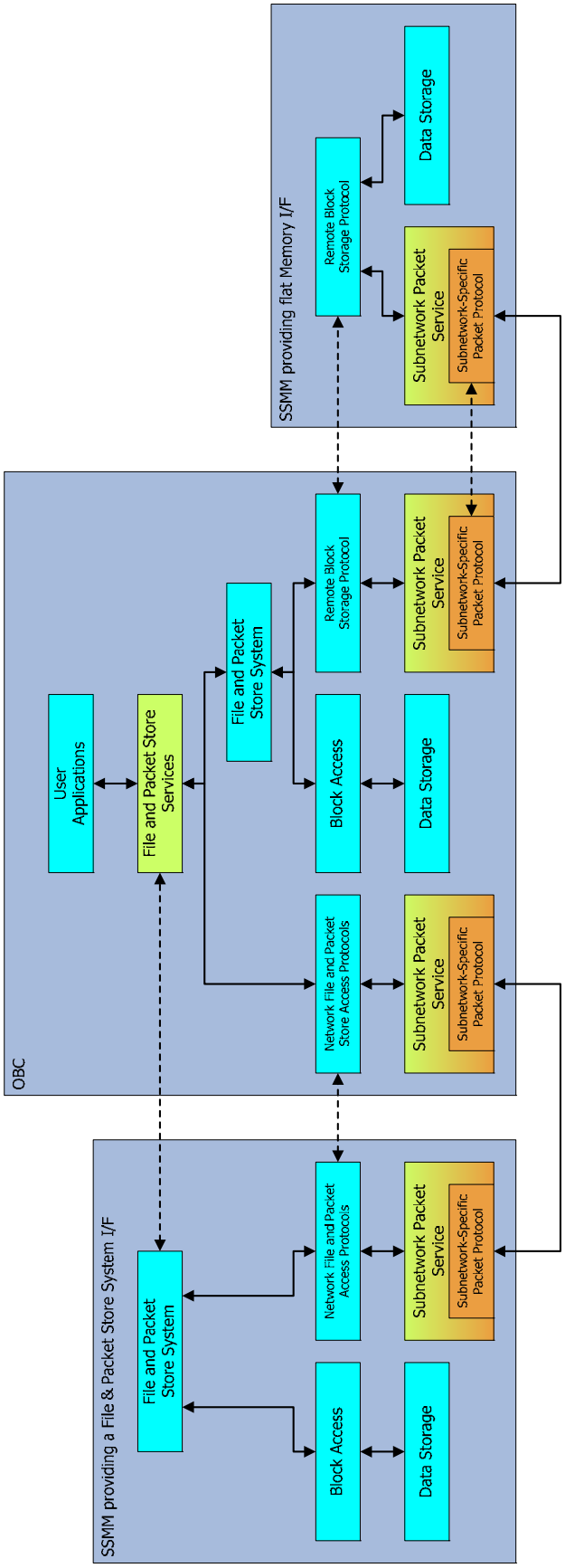

Figure 2-4: Example Deployment of FPSS and File and Packet Store Systems

© ISO 2015 – All rights reserved

AVA Documents
Document Sharing Hub

Full document details: BS ISO 20217:2015

2.5 EXAMPLE DEPLOYMENT

There are a variety of possible provisions of file and packet store systems and the associated access from the FPSS running on an OnBoard Computer (OBC), illustrated in figure 2-4:

a) a local file and packet store system using:

   1) local data storage, directly accessed;

   2) remote data storage on a simple mass memory ('SSMM[1] providing flat Memory I/F'), accessed using a Remote Block Storage Protocol;

b) a remote file and packet stores on a complex mass memory ('SSMM providing a File System I/F'), accessed using Network File and Packet Access Protocols.

It should be noted that, in addition, a file and/or packet store can be composed of more than one mass memory device interfaced by a single file or packet store system. Multiple, distributed mass memories may be accessed through the single file or packet store system over the same subnetwork or different subnetworks (see figure 2-5, below). This requires the FPSS to use protocols providing network file access and packet store access (labelled generically as Network File and Packet Access Protocols). However, any protocol used specifically by the file or packet store system (labelled generically as Remote Block Storage Protocol, not specified here) is expected to be distinct from the protocols used by the FPSS. The application is isolated from any use of such multiple, distributed mass memories; e.g., file and directory names need to be unique across a complete file store, not just within an individual mass memory.

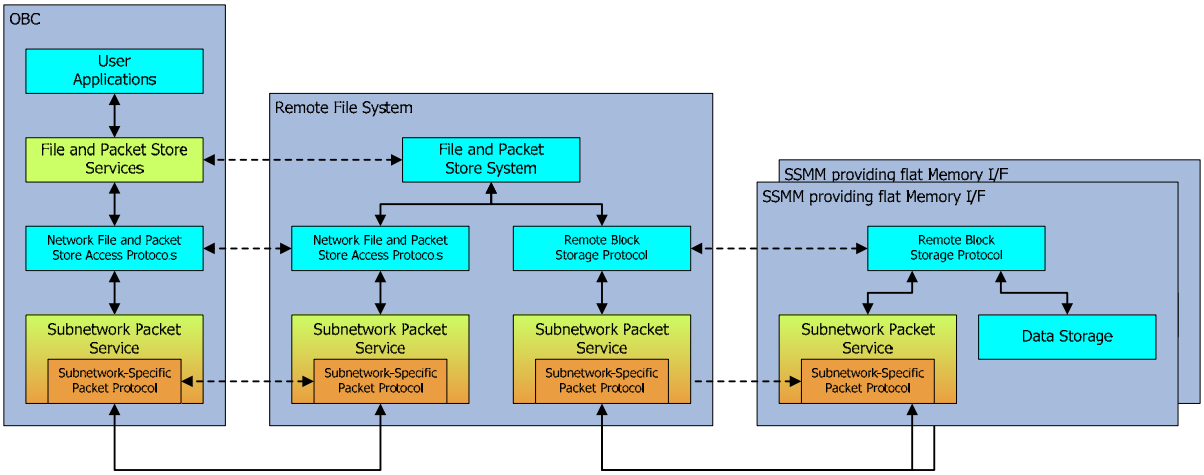

Figure 2-5: Multiple Distributed Mass Memories in a File Store

[1] SSMM – Solid State Mass Memory

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

With regard to the dumping of telemetry packets, as described in 2.3.4, the interface between a Packet Store and the Spacecraft Telemetry System required for this function is private and outside the scope of the Packet Store Services. Below are illustrations providing two possible implementations, figure 2-6 for a mass memory providing a Packet Store Service interface, and figure 2-7 for a Packet Store System implemented on an onboard computer.

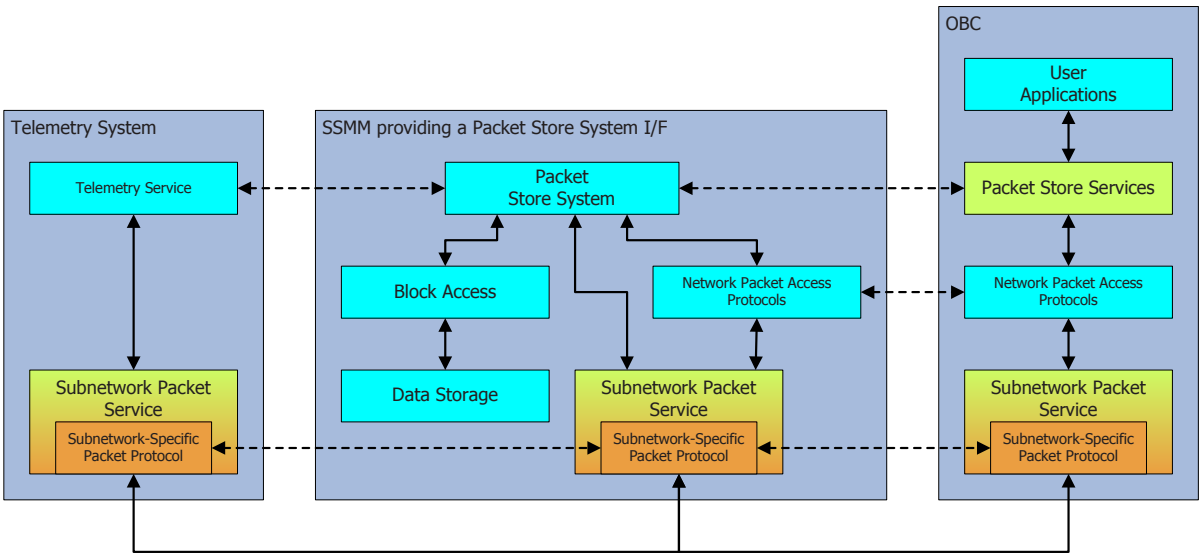

Figure 2-6: Example Interfacing from SSMM-based Packet Store System to Telemetry System

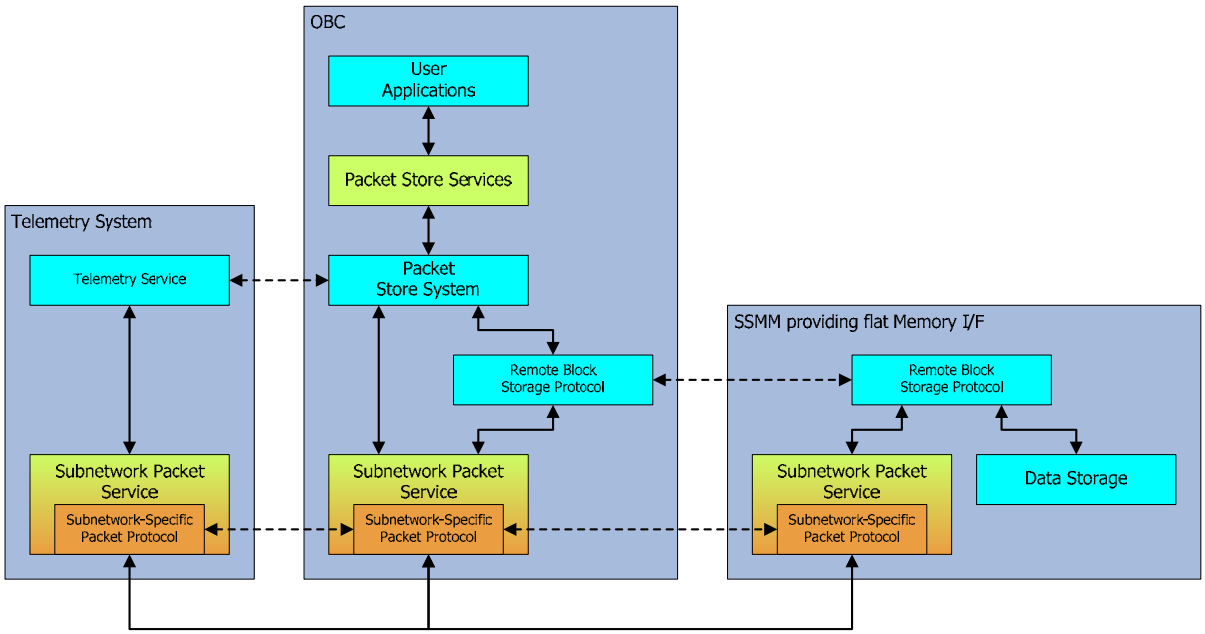

Figure 2-7: Example Interfacing from OBC-based Packet Store System to Telemetry System

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

3 SERVICE DEFINITION

3.1 PROVIDED SERVICE

3.1.1 GENERAL

NOTE – The FPSS are divided into the following four services:

- File Access Service;
- File Management Service;
- Packet Store Access Service;
- Packet Store Management Service.

3.1.1.1 To provide a file system onboard a spacecraft, both the File Access Service and the File Management Service shall be provided.

3.1.1.2 To provide a packet store onboard a spacecraft, both the Packet Store Access Service and the Packet Store Management Service shall be provided.

3.1.1.3 A packet store may be provided where only a file system is required.

3.1.1.4 A file system may be provided where only a packet store is required.

NOTE – The following subsections detail the provisions of each of these services.

3.1.2 FILE ACCESS SERVICE

3.1.2.1 The *File Access Service* (FAS) shall enable a user entity to access 'existing' files in a file store, regardless of that file store's location; i.e., the accessed files can reside on local or remote file stores with respect to the user entity.

3.1.2.2 The service shall, as a minimum, give a user entity the ability to:

- open and close a file;
- read from a file;
- write to a file.

3.1.2.3 Optionally, the service may give a user entity the ability to 'seek', i.e., move, the user entity's current position within a file.

3.1.2.4 Each FAS implementation shall support either a local file store or a remote file store and may support both.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

3.1.2.5 If a local file store is implemented, each FAS implementation shall be mapped to a single, local file store's file system.

NOTE – For a local file store, whilst the file system itself is local, the associated storage may be remote, in which case the storage is accessed by a Remote Block Storage Protocol (not defined here).

3.1.2.6 If a remote file store is implemented, each FAS implementation shall be mapped onto a Network File Access Protocol (see 3.2.4).

3.1.2.7 Such mappings shall be transparent to a user entity.

3.1.3 FILE MANAGEMENT SERVICE

3.1.3.1 General

3.1.3.1.1 The *File Management Service* (FMS) shall enable a user entity to create and manipulate files in a file store, regardless of that file store's location; i.e., the accessed files can reside on local or remote file stores with respect to the user entity.

3.1.3.1.2 The service shall, at a minimum, give a user entity the ability to:

- list the contents of a directory;
- copy and move files;
- create and delete files.

3.1.3.1.3 Optionally, the service may give a user entity the ability to:

- create, rename, and delete directories;
- get and change the user entity's current directory;
- rename files;
- lock or unlock files;
- list the currently locked files;
- find a file within a file store;
- get the status of a file.

3.1.3.1.4 To manage a file store, each FMS implementation must be mapped to a file store's file system.

3.1.3.1.5 Such mappings shall be transparent to a user entity.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

3.1.3.2 Locking Files

3.1.3.2.1 If file locking is implemented, a user entity shall be able to lock files in order to prevent other users from reading from or writing to a particular file.

3.1.3.2.2 The user entity that applied a lock to a file shall be known as the lock-owner.

3.1.3.2.3 A file shall stay locked until it is released by a user entity (not necessarily the lock owner).

NOTE – Locks provide control of access to files.

3.1.3.2.4 The access control options associated with locks are: no access, read-only access, and read-write access. Different access control associated with locks can be applied to lock-owners compared with all other user entities. The following permutations are supported:

- lock-owner read-only access, all other user entities no access;
- lock-owner read-only access, all other user entities read-only access;
- lock-owner read-write access, all other user entities no access.

3.1.3.2.5 In case of successful locking, an identifier shall be associated with the lock, known as the lock identifier, and the lock shall remain until unlocked.

3.1.3.2.6 Any user entity providing the correct lock identifier shall be able to unlock the locked file.

3.1.3.2.7 A user entity shall be able to obtain a list of current lock identifiers.

NOTE – This, for example, may be used to obtain the locks held by a failed application so that they may be unlocked.

3.1.4 PACKET STORE ACCESS SERVICE

3.1.4.1 The *Packet Store Access Service* (PSAS) shall enable a user entity to access packets in a packet store, regardless of that packet store's location; i.e., the accessed packets can reside on local or remote packet stores with respect to the user entity.

3.1.4.2 The packet store may provide a private interface to the spacecraft telemetry system to permit packets to be dumped (i.e., sent) directly from the packet store to that system for transmission without being transferred over the interface to the PSAS implementation.

3.1.4.3 The service shall, at a minimum, give a user entity the ability to:

- get list of all packet stores;
- clear all packets from a packet store;

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

– write packets to a packet store;

– read packets from a packet store;

– set the read pointer position in a packet store;

– delete packets from a packet store;

– get the status of a packet store.

3.1.4.4 Optionally, the service may give a user entity the ability to read and free selected packets in a packet store (random-access packet stores only).

3.1.4.5 Optionally, if the packet store implements a private interface to the spacecraft telemetry system, the service may give a user entity the ability to:

– dump packets to the telemetry system;

– dump to the telemetry system selected packets in a packet store (random-access packet stores only).

3.1.4.5.1 To access a local packet store, each PSAS implementation must be mapped onto a single local packet store's storage system.

NOTE – For a local packet store, whilst the storage system itself is local, the associated storage may be remote, in which case the storage is accessed by a Remote Block Storage Protocol (not defined here).

3.1.4.5.2 To access a remote packet store, each PSAS implementation must be mapped onto a Network Packet Access Protocol (see 3.2.5). Such mappings shall be transparent to the user entity.

3.1.5 PACKET STORE MANAGEMENT SERVICE

3.1.5.1 The *Packet Store Management Service* (PSMS) shall enable a user entity to manage packet stores, regardless of that packet store's location; i.e., it can be local or remote with respect to the user entity on the same spacecraft.

3.1.5.2 The service shall at a minimum give a user entity the ability to create and delete packet stores.

3.2 SERVICES EXPECTED FROM THE UNDERLYING LAYERS

3.2.1 OVERVIEW

The services that the FPSS expect from the underlying layers are divided into the following capabilities:

– local file store services;

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

– local packet store services;

– network file access protocol (optional);

– network packet access protocol (optional).

The requirements on each are detailed in the following subsections.

3.2.2 LOCAL FILE STORE SERVICES

NOTE – As the way in which the storage capability is provided may vary, the FPSS is built on the premise that any file or organized set of files (i.e., a file store) can be accessed and managed through a set of file system services.

3.2.2.1 The file store shall provide either of the following file system types:

– flat file system;

– hierarchical file system.

3.2.2.2 The file store shall provide the following set of file system services:

– list contents of directory;

– create file;

– open a file;

– close a file;

– read from file;

– write to file;

– delete file;

– move file;

– copy file.

3.2.2.3 Optionally, the file store may provide the following set of file system services:

– create directory;

– remove directory;

– rename directory;

– get user entity's current directory;

– change user entity's current directory;

– lock and unlock files;

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

– move a user entity's current position within file;

– get file status.

3.2.2.4 The file store shall maintain the following attributes to a file:

– name;

– lock status (only where the lock and unlock files capabilities are provided);

– file size.

3.2.2.5 Optionally, the file store may maintain the following attributes to a file:

– creation time;

– last write time.

3.2.2.6 In an implementation of the FPSS, the file system services must be mapped to and from actual hardware and software that constitute the real file store.

3.2.2.7 These file system services shall be used by the FPSS to access, manage, and transfer files in a local file store or via a network file access protocol (see 3.2.4) for a remote file store.

3.2.3 LOCAL PACKET STORE SERVICES

NOTE – As the way in which the storage capability is provided may vary, the FPSS is built on the premise that any packet or organized set of packets (i.e., a packet store) can be accessed and managed through packet storage system services.

3.2.3.1 The packet store shall provide the following packet store types:

– bounded FIFO packet store.

3.2.3.2 The packet store may provide the following packet store types:

– circular FIFO packet store.

– hierarchical file system;

3.2.3.3 The packet store shall provide the following set of packet storage system services:

– reset—delete all packets in a packet store;

– write—write packets to the packet store;

– read—read the next *n* packets from the packet store;

– move by—move the user's next packet pointer backwards or forwards by *n* packets in the packet store;

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

– free—free the oldest *n* packets in the packet store;

– status report—report the status of the packet store;

3.2.3.4 Optionally, the packet store may provide the following set of packet storage system services:

– dump—dump packets to the ground using the spacecraft telemetry system;

– selective seek—create a new time-ordered set of packets in the packet store (random-access packet stores only) meeting the specified selection criteria;

– selective read—read the next *n* packets from the time-ordered set in the packet store (random-access packet stores only) selected by the selective seek capability;

– selective free—free the oldest *n* packets in the time-ordered set in the packet store (random-access packet stores only) selected by the selective seek capability;

– selective dump—dump the next *n* packets from the time-ordered set in the packet store (random-access packet stores only) selected by the selective seek capability.

3.2.3.5 The Packet Store shall support (where provided) concurrently the following operations: reading, selective reading, writing, dumping, freeing, and selective freeing of disjoint packets in the same packet store.

3.2.3.6 In an implementation of the FPSS, the packet storage system services must be mapped to and from actual hardware and software that constitute the real packet store.

3.2.3.7 These packet storage system services shall be used by the FPSS to access packets in a local packet store or via a network packet access protocol for a remote packet store.

3.2.4 NETWORK FILE ACCESS PROTOCOL

Where support is provided for the set of capabilities defined in 3.2.2 for accessing and managing files residing in a *remote* file store, an underlying protocol providing network file access is required to be used. This specification does not include a definition of the functions or design of such a protocol, this being a matter of the FPSS implementation.

NOTE – As the way in which the storage capability is provided may vary, the FPSS is built on the premise that any file or organized set of files (i.e., a file store) can be accessed and managed remotely through a network file system interface protocol.

3.2.5 NETWORK PACKET ACCESS PROTOCOL

Where support is provided for the set of capabilities defined in 3.2.3 for accessing and managing packets residing in a *remote* packet store, an underlying protocol providing

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

network packet access is required to be used. This specification does not include a definition of the functions or design of such a protocol, this being a matter of the FPSS implementation.

NOTE – As the way in which the storage capability is provided may vary, the FPSS is built on the premise that any packet or organized set of packets (i.e., a packet store) can be accessed and managed remotely through a network packet access protocol.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

3.3 SERVICE PARAMETERS

3.3.1 GENERAL

The File and Packet Store Services shall use the parameters specified in 3.3.2 to 3.3.4.

3.3.2 GENERAL PARAMETERS

3.3.2.1 Transaction Identifier

3.3.2.1.1 The Transaction Identifier parameter shall be a value, assigned by the invoking user entity, which is subsequently used to associate indication primitives with the causal request primitives.

NOTE – The user entity is thus able to correlate all indications and confirmations with the originating service request.

3.3.2.1.2 The Transaction Identifier shall be unique within the user application entity.

3.3.2.1.3 Uniqueness in the service provider shall be achieved through the concatenation of Transaction Identifier and either Packet Store Access Service Access Point (PSASAP) Address or Packet Store Management Service Access Point (PSMSAP) Address depending upon the service.

3.3.2.2 Result Metadata

The Result Metadata parameter shall be used to provide information generated by the service provider to the user entity to provide information related to the successful or failed result of a device access operation.

NOTE – The parameter can also include other information indicating failure conditions; e.g., the specified request could not be serviced within the managed timeout period or the Device Access Service is not functioning correctly.

3.3.3 FILE ACCESS AND MANAGEMENT SERVICE PARAMETERS

3.3.3.1 File Identifier

The File Identifier parameter shall be used to logically identify an opened file in a file store.

3.3.3.2 File Name

The File Name parameter shall be a string used to uniquely identify a file in a directory and may or may not contain the full file path from the ‘root’ directory (see 3.3.3.6). If no path is provided, it shall be assumed to refer to the user entity’s current directory.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

3.3.3.3 File Attributes

The File Attributes parameter shall be used to list the attributes of a file, containing the following information:

- **File Name** identifies the file (see 3.3.3.2).
- **Creation Time** (optionally) is the time at which the file was created.
- **Last Write Time** (optionally) is the time at which the file was last written to.
- **Lock Identifier** (optionally) identifies a lock on the file (if locked) (see 3.3.3.12).
- **Lock Type** identifies the type of lock on the file (if locked) (see 3.3.3.11).
- **Lock Owner** identifies the user entity owning the lock on the file (if locked).
- **Size** is the size of the file in octets.

3.3.3.4 Directory Listing

The Directory Listing shall be used to list the attributes (see 3.3.3.3) of all files in a directory.

3.3.3.5 File Selection Pattern

The File Selection Pattern parameter shall be used to specify the pattern of any single file attribute (see 3.3.3.3) on which the files are to be found. This may include wildcards and string pattern matching expressions.

NOTE – The format and implementation of wildcards and string pattern matching expressions is implementation-specific.

3.3.3.6 Directory Name

3.3.3.6.1 The Directory Name parameter shall be used to uniquely identify a directory and may or may not contain the full path from the ‘root’ directory.

3.3.3.6.2 If no path is provided, it shall be assumed to be in the user entity’s current directory.

3.3.3.6.3 A special Directory Name parameter value shall identify the ‘root’ directory.

3.3.3.6.4 A special Directory Name parameter value may identify the directory above the user entity’s current directory in the hierarchy.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

3.3.3.7 File Opening Criteria

3.3.3.7.1 The File Opening Criteria parameter shall indicate the criteria to be applied to a file when opened. It should be noted that the access rights granted to the file may differ from the ones requested by the File Opening Criteria parameter. Possible attributes associated with opening a file are:

- **Access Type**:
  - *Read-only*—the user entity can only read from the file;
  - *Read-write*—the user entity can read from or write to the file.
- **Create**—if specified, the file does not exist it will be created.
- **Lock Type**—(optionally) if specified, the file is locked atomically when opened (see 3.3.3.11).
- **Append**—if specified, the user entity's current position is initialized to the end of the file.

3.3.3.7.2 In the above list of attribute types only the Access-type is mandatory.

3.3.3.8 File Offset

The File Offset parameter shall indicate the desired current position within a file to which file data is to be read from or written to. Offset 'zero' indicates the first octet in the file.

3.3.3.9 File Data Length

The File Data Length parameter shall indicate the length in octets of the data to be read or written (including the octet located at the current position).

3.3.3.10 File Data

The File Data parameter shall be the data read from or to be written to a file.

3.3.3.11 Lock Type

The Lock Type parameter shall specify a lock action on a file. Possible values are:

- *Exclusive_Read_Only*. The lock-owner can only read from the file. Other user entities cannot read from or modify the file.
- *Read_Only*. The lock-owner can only read from the file. Other user entities can only read from the file.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015

– *Exclusive_Access*. The lock-owner can read from and modify the file. Other user entity cannot read from or modify the file.

3.3.3.12 File Lock Identifier

The File Lock Identifier parameter shall be used to logically identify an open lock on a file.

3.3.3.13 File Lock List

The File Lock List parameter shall be used to list currently locked files and shall consist of entries, each containing the following information:

– **File Lock Identifier** identifies a lock on a file (see 3.3.3.12).

– **File Name** identifies the associated file upon which the lock is applied (see 3.3.3.2).

– **Lock Type** identifies the associated lock action (see 3.3.3.11).

– **Lock Owner** identifies the associated user entity owning the lock.

3.3.4 PACKET STORE ACCESS AND MANAGEMENT SERVICE PARAMETERS

3.3.4.1 Packet Store Description List

The Packet Store Description List parameter shall be used to list descriptions of packet stores. Each packet store description consists of the following information:

– **Packet Store Identifier** identifies the packet store. This is unique within the spacecraft domain.

– **Packet Store Type** is the type of packet store; either Bounded FIFO, Circular FIFO or Random-Access.

– **Packet Type** is a flag indicating the type of packets in the packet store: either CCSDS or non-CCSDS packets.

– **Number of Stored Packets** is the number of packets currently stored in a packet store.

– **Time Range** is the time of the earliest packet and the time of the latest packet in the packet store.

– **Allocated Space** is the amount of octets allocated to storage of packets in the packet store.

– **Free Space** is the number of octets currently not used to store packets in the packet store.

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 20217:2015