

DRAFT Recommendation for Space Data System Standards

|  |
| --- |
|  |

DRAFT

Red Book

Month Year

Dedication

AUTHORITY

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  | Issue: | Red Book, Issue 1 |  |
|  | Date: | Month Year |  |
|  | Location: | Washington, DC, USA |  |
|  |  |  |  |

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

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and represents the consensus technical agreement of the participating CCSDS Member Agencies. The procedure for review and authorization of CCSDS documents is detailed in *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4), and the record of Agency participation in the authorization of this document can be obtained from the CCSDS Secretariat at the e-mail address below.

This document is published and maintained by:

CCSDS Secretariat

National Aeronautics and Space Administration

Washington, DC, USA

E-mail: secretariat@mailman.ccsds.org

STATEMENT OF INTENT

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

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

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

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

o Whenever a member establishes a CCSDS-related **standard**, that member will provide other CCSDS members with the following information:

-- The **standard** itself.

-- The anticipated date of initial operational capability.

-- The anticipated duration of operational service.

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

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

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

**PREFACE**

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

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

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

FOREWORD

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CCSDS has processes for identifying patent issues and for securing from the patent holder agreement that all licensing policies are reasonable and non-discriminatory. However, CCSDS does not have a patent law staff, and CCSDS shall not be held responsible for identifying any or all such patent rights.

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Recommended Standard is therefore subject to CCSDS document management and change control procedures, which are defined in *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4). Current versions of CCSDS documents are maintained at the CCSDS Web site:

http://www.ccsds.org/

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

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

Member Agencies

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

Observer Agencies

* Austrian Space Agency (ASA)/Austria.
* Belgian Federal Science Policy Office (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.
* Danish National Space Center (DNSC)/Denmark.
* Departamento de Ciência e Tecnologia Aeroespacial (DCTA)/Brazil.
* Electronics and Telecommunications Research Institute (ETRI)/Korea.
* European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
* European Telecommunications Satellite Organization (EUTELSAT)/Europe.
* Geo-Informatics and Space Technology Development Agency (GISTDA)/Thailand.
* Hellenic National Space Committee (HNSC)/Greece.
* Indian Space Research Organization (ISRO)/India.
* Institute of Space Research (IKI)/Russian Federation.
* 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.
* South African National Space Agency (SANSA)/Republic of South Africa.
* Space and Upper Atmosphere Research Commission (SUPARCO)/Pakistan.
* Swedish Space Corporation (SSC)/Sweden.
* Swiss Space Office (SSO)/Switzerland.
* United States Geological Survey (USGS)/USA.

**DOCUMENT CONTROL**

|  |  |  |  |
| --- | --- | --- | --- |
| **Document** | **Title** | **Date** | **Status** |
|  | , Draft Recommended Standard, Issue 2 | Month  Year | Current Draft |
|  |  |  |  |
|  |  |  |  |

**CONTENTS**

[**1.** **INTRODUCTION** 9](#_Toc88481833)

[**1.1** **PURPOSE** 9](#_Toc88481834)

[**1.2** **SCOPE** 9](#_Toc88481835)

[**1.3** **ORGANIZATION OF THE RECOMMENDED STANDARD** 9](#_Toc88481836)

[**1.4** **DEFINITIONS** 10](#_Toc88481837)

[**1.5** **REFERENCES** 12](#_Toc88481838)

[**2.** **OVERVIEW** 13](#_Toc88481839)

[**2.1** **GENERAL** 13](#_Toc88481840)

[**2.2** **IMPLEMENTATION ARCHITECTURES** 15](#_Toc88481841)

[**2.3** **SERVICES PROVIDED BY BP** 15](#_Toc88481842)

[**2.4** **QUALITIES OF SERVICE NOT PROVIDED BY BP** 15](#_Toc88481843)

[**2.5** **SIGNIFICANT CHANGES FROM CCSDS 734.2-B-1** 16](#_Toc88481844)

[**3.** **CCSDS PROFILE OF RFC 9171** 17](#_Toc88481845)

[**3.1** **GENERAL** 17](#_Toc88481846)

[**3.3** **IPN NAMING SCHEME** 17](#_Toc88481847)

[**3.5** **USE OF TIME IN SECTION 4.2.6 OF RFC 9171** 18](#_Toc88481848)

[**4** **SERVICE DESCRIPTION** 18](#_Toc88481849)

[**4.1** **SERVICES AT THE USER INTERFACE** 18](#_Toc88481850)

[**4.2** **SUMMARY OF PRIMITIVES** 19](#_Toc88481851)

[**4.3** **SUMMARY OF PARAMETERS** 19](#_Toc88481852)

[**4.4** **BP SERVICE PRIMITIVES** 21](#_Toc88481853)

[**5** **SERVICES BP REQUIRES OF THE SYSTEM** 27](#_Toc88481854)

[**5.1** **RELIABLE STORAGE REQUIREMENTS** 27](#_Toc88481855)

[**5.2** **UNDERLYING COMMUNICATION SERVICE REQUIREMENTS** 27](#_Toc88481856)

[**6** **CONFORMANCE REQUIREMENTS** 29](#_Toc88481857)

[**6.1** **General Requirements** 29](#_Toc88481858)

[**6.1.1** **Protocol Implementation** 29](#_Toc88481859)

[**6.2** **BUNDLE PROTOCOL REQUIREMENTS** 29](#_Toc88481860)

[**ANNEX A Protocol Implementation Conformance Statement Proforma (Normative)** 31](#_Toc88481861)

[**A3** **Notation** 32](#_Toc88481862)

[**A4** **Referenced Base Standards** 32](#_Toc88481863)

[A5 General Information 33](#_Toc88481864)

[**ANNEX B Convergence LAyer aDapters (Normative)** 38](#_Toc88481865)

[**B1** **Overview** 38](#_Toc88481866)

[**B2** **Convergence Layer Adapters** 38](#_Toc88481867)

[B3 TCP convergence Layer adapter 38](#_Toc88481868)

[B4 UDP convergence Layer adapter – Encapsulation of bundles in udP datagrams 38](#_Toc88481869)

[B5 LTP convergence Layer adapter 39](#_Toc88481870)

[**B6** **CCSDS ENCAPSULATION SERVICE CONVERGENCE LAYER ADAPTER—ENCAPSULATION OF BUNDLES VIA THE CCSDS ENCAPSULATION SERVICE** 40](#_Toc88481871)

[**B7** **USLP CONVERGENCE LAYER ADAPTER** 40](#_Toc88481872)

[**B8** **SPACE PACKETS** 40](#_Toc88481873)

[**Annex C AGGREGATE CUSTODY SIGNAL Specification (Normative)** 41](#_Toc88481874)

[**C1** **OVERVIEW** 41](#_Toc88481875)

[**C2** **DEFINITION OF TERMS** 41](#_Toc88481876)

[**C3** **ACS COMPONENT FORMATS** 42](#_Toc88481877)

[**C4** **ACS BEHAVIOR** 43](#_Toc88481878)

[**ANNEX E** 48](#_Toc88481879)

[**E1** **BASIC REQUIREMENTS** 48](#_Toc88481880)

[**ANNEX F** 53](#_Toc88481881)

[**ANNEX G** 55](#_Toc88481882)

[**SANA Registry Considerations** 55](#_Toc88481883)

[ANNEX H 56](#_Toc88481884)

[ANNEX I 57](#_Toc88481885)

[**ANNEX J** 57](#_Toc88481886)

[**ANNEX K** 58](#_Toc88481887)

# **1. INTRODUCTION**

## **1.1 PURPOSE**

The purpose of this document is to establish a recommended standard for CCSDS Bundle Protocol (BP), based on the bundle protocol of RFC 9171 (reference [1]), which defines end-to-end protocol, bundle structure, block types, and abstract service descriptions for the exchange of messages (bundles) that support Delay Tolerant Networking (DTN). BP provides Application Layer service to networks allowing them to utilize BP’s capabilities:

* ~~Ability to use physical motility for the movement of data~~.
* Ability to move the responsibility for error control from one node to another
* Ability to cope with intermittent connectivity, including cases where the sender and receiver are not concurrently present in the network
* Ability to take advantage of scheduled, predicted, and opportunistic connectivity, whether bidirectional or unidirectional, in addition to continuous connectivity
* Ability to use available bandwidth for a wide variety of services and functions
* Late binding of overlay network endpoint identifiers to underlying constituent network addresses

## **1.2 SCOPE**

This Recommended Standard is designed to be applicable to any kind of space mission infrastructure that requires delay and disruption tolerance, regardless of resources, latencies, or complexity. It is intended that this Recommended Standard become a uniform standard among all CCSDS Agencies. In addition, this specification exists to utilize the underlying service of various interworking protocols both onboard and in transit between ground and space-based assets.

This Recommended Standard is intended to be applied to all systems that claim conformance to the CCSDS Bundle Protocol version 7. For users who desire to use BP version 6 for their implementation, please refer to CCSDS spec 734.2-B-1. BP is agnostic to the choice of underlying transmission protocol in that BP can function over AOS, USLP, Encapsulation Packet Protocol, Space Packet, Proximity-1 Space Link Protocol, and various Internet and ground-based protocols.

The CCSDS believes it is important to document the rational underlying the recommendations chosen, so that future evaluations of proposed changes or improvements will not lose sight of previous decisions. The concept and rationale for the use of a bundles protocol in space links may be found in reference [H1].

## **1.3 ORGANIZATION OF THE RECOMMENDED STANDARD**

This recommended Standard is organized as follows:

* Section 2 contains an overview of the Bundle Protocol and the references from which it is derived.
* Section 3 contains the CCSDS modification to RFC 9171.
* Section 4 contains the service descriptions.
* Section 5 contains services BP requires of the system.
* Section 6 contains conformance requirements.
* Annex A contains the Implementation Conformance Statement for the protocol.
* Annex B contains the Convergence Layer Adapters (CLAs).
* Annex C contains the Aggregate Custody Signal specification
* Annex D contains the Bundle-In-Bundle Encapsulation
* Annex E contains BP managed information
* Annex F contains Security, Space Assigned Numbers Authority (SANA), and Patent considerations.
* Annex G contains Informative References.
* Annex H contains abbreviations and acronyms used in this document.

## **1.4 DEFINITIONS**

**1.4.1 Definitions From Open Systems Interconnection (OSI) Service Definition Conventions**

This Recommended Standard makes use of a number of terms defined in reference [3]. As used in this Recommended Standard those terms are to be interpreted in a generic sense, i.e., in the sense that those terms are generally applicable to any of a variety of technologies that provide for the exchange of information between real systems. Those terms are:

* Indication;
* Primitive;
* Request;
* Response;

**1.4.2 Definitions From OSI Basic Reference Model**

This Recommended Standard makes use of a number of terms defined in reference [4]. As used in this Recommended Standard those terms are to be understood in a generic sense, i.e., in the sense that those terms are generally applicable to any of a variety of technologies that provide for the exchange of information between real systems. Those terms are:

* Entity;
* Protocol Data Unit (PDU);
* Service;
* Service Data Unit (SDU);

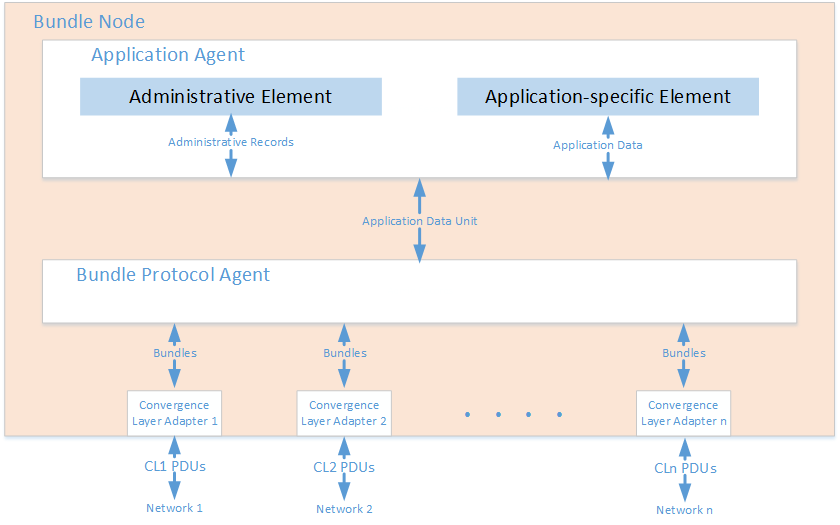
**1.4.3 Definitions From RFC 9171**

***1.4.3.1 Overview***

This Recommended Standard makes use of a number of terms defined in reference [1]. Some of the definitions needed for section 2 of this document are reproduced here for convenience.

A graphical representation of a bundle node is given in figure 1-1. A bundle node is any entity that can send and /or receive bundles.

Each bundle node has three conceptual components described in more detail below: a ‘bundle protocol agent’, a set of zero or more ‘convergence layer adapters’, and an ‘application agent’. The major components are illustrated in figure 1-1 (“CL1 PDUs” are the PDUs of the convergence-layer protocol used in network).



**Figure 1-1: Graphical Representation of a Bundle Node**

It should be noted that there is *one* application agent per conceptual bundle node. That application agent may provide communication services to multiple applications and register in multiple endpoints (may provide multiple endpoint identifiers to the bundle protocol agent, requesting delivery of bundles to any of those endpoints).

#### **1.4.3.2 RFC 9171 Terms**

**Bundle**: A bundle is a protocol data unit of BP, so named because negotiation of the parameters of a data exchange may be impractical in a delay-tolerant network: it is often better practice to “bundle” with a unit of application data all metadata that might be needed in order to make the data immediately usable when delivered to the application. Each bundle comprises a sequence of two or more “blocks” of protocol data, which serve various purposes.

**Bundle Node**: A bundle node (or, in the context of this document, simply a “node”) is any entity that can send and/or receive bundles. Each bundle node has three conceptual components: a “bundle protocol agent”, a set of zero or more “convergence layer adapters”, and an “application agent”.

**Bundle Protocol Agent**: The bundle protocol agent (BPA) of a node is the node component that offers the BP services and executes the procedures of the bundle protocol.

**Application Agent**: The application agent (AA) of a node is the node component that utilizes the BP services to effect communication for some user purpose. The application agent in turn has two elements, an administrative element and an application-specific element.

**Administrative Element**: Administrative element - The administrative element of an AA is the node component that constructs and requests transmission of administrative records (defined below), including status reports, and accepts delivery of and processes any administrative records that the node receives.

**Application-Specific Element**: Administrative element - The administrative element of an AA is the node component that constructs and requests transmission of administrative records (defined below), including status reports, and accepts delivery of and processes any administrative records that the node receives.

**Convergence Layer Adapter**: A convergence layer adapter (CLA) is a node component that sends and receives bundles on behalf of the BPA, utilizing the services of some 'native' protocol stack that is supported in one of the networks within which the node is functionally located.

## **1.5 REFERENCES**

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

1. S. Burleigh, K. Fall, and E. Birrane. *Bundle Protocol Version*. RFC xxx. Reston, Virgina: ISOC, January 25, 2021.
2. K. Scott and S. Burleigh. *Bundle Protocol Specification*. RFC 5050. Reston, Virginia: ISOC, November 2007.
3. *Information Technology—Open Systems Interconnection—Basic Reference Model—Conventions for the Definition of OSI Services*. International Standard, ISO/IEC 10731:1994. Geneva: ISO, 1994.
4. *Information Technology—Open Systems Interconnection—Basic Reference Model: The Basic Model*. 2nd ed. International Standard, ISO/IEC 7498-1:1994. Geneva: ISO, 1994.
5. S. Burleigh. *Compressed Bundle Header Encoding (CBHE)*. RFC 6260. Reston, Virginia: ISOC, May 2011.
6. S. Burleigh. *CBHE-Compatible Bundle Multicast*. RFC xxx. Restion, Virginia: ISOC, November 7, 2012.
7. Space Assigned Numbers Authority (SANA). <http://sanaregistry.org/>.
8. L. Eggert and G. Fairhurst. *Unicast UDP Usage Guidelines for Application Designers*. RFC 5405. Reston, Virginia: ISOC, November 2008.
9. Keith Scott’s new LTP book
10. *Encapsulation Packet Protocol*. Issue 3. Recommendation for Space Data System Standards (Blue Book), CCSDS 133.1-B-3. Washington, D.C.: CCSDS, May 2020.
11. M. Ramadas, S. Burleigh, and S. Farrell. *Licklider Transmission Protocol—Specification*. RFC 5326. Reston, Virginia: ISOC, September 2008.

# **2. OVERVIEW**

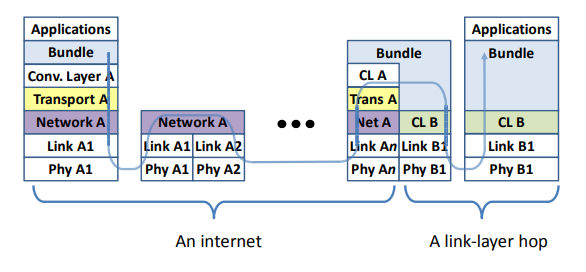
## **2.1 GENERAL**

Delay Tolerant Networking is an end-to-end network service providing communications in and/or through environments characterized by one or more of the following:

* Intermittent connectivity;
* Variable delays, which may be large and irregular;
* Intermittent transmission error rates;
* Asymmetric and simplex links.
* Disparate Data Rates

One core element of DTN is the BP. BP provides end-to-end network services, operating above the data transport services provided by links or networks accessed via the CLAs, and forming a store-and-forward network.

BP uses underlying “native” transport and/or network protocols for communications within a given constituent network. The layer at which those underlying protocols lie is known as the “convergence layer”. The interface between the BP layer and the convergence layer is known as the “convergence layer adapter”. This concept is illustrated in Figure 2-1. PDUs traveling from the application and bundle layer encounter a CLA, which is responsible for sending (and receiving) bundles according to the “native” protocol that the convergence layer uses underneath it (as interpreted in a standard OSI model with BP additions). Typically, a specific CLA is created for each unique “native” protocol. The CLA on the left, for example, could represent an adapter specific to a TCP network. The CLA on the right (CL B) could represent an interface to the Licklider Transmission Protocol (LTP), with “Link B1” representing LTP running over a CCSDS Data Link Layer protocol. Alternatively, BP can be used to support a connection between two separate internets e.g. an on-orbit internet and a ground internet, terrestrial or otherwise.



**Application Layer**

**Figure 2.1: Bundle Protocol End-to-End Delivery Service**

This document describes the format of the messages (called bundles) passed between nodes participating in bundle transmission. In addition, this document addresses endpoint naming for the purpose of bundle header compression and describes how the protocol may be extended to support new capabilities while maintaining compatibility with the base protocol. This document does not address bundle routing algorithms (e.g. SABR), mechanisms for populating the routing or forwarding information bases of bundle nodes, nor methods for scheduling bundle transmission (e.g. Contact Plan).

General refactoring of the bundle protocol has improved the protocol in terms of simplicity, power, and flexibility since the protocol was first released for industry purposes in CCSDS 734.2-B-1. This offers the improved bundle protocol increased scalability and compatibility from its previous iteration. Additionally, the bundle protocol directly supported by CCSDS 734.2-B-1 will no longer receive support once BPbis is released for use in industry. Therefore, this document, upon publication, will obsolete CCSDS 734.2-B-1.

The bundle protocol discussed in BPbis RFC should not be considered for deployment on the global internet. It has improved scalability compared to its previous iteration discussed in CCSDS 734.2-B-1, but still would experience issues such as scalability to millions of nodes, congestion control, and non-destructive coexistence with other established protocols (particularly the Transmission Control Protocol [TCP]). However, bundle protocol is intended for harsh environments where more commonly known communications protocols (e.g. TCP) tend to break down and stop functioning. In scenarios like these, bundle protocol is an excellent technological innovation that allows multiple internetworking environments in previously unconnected locations to interact.

The SIS-DTN working group has carefully considered the protocol specified in the BPbis RFC and has determined that it is suitable for adoption, together with the modifications in section 3 of this document, for use in CCSDS missions. In particular, CCSDS missions do not have the same scalability issues as the Internet, and testing has demonstrated that the profile defined in this document is suitable for CCSDS environments.

## **2.2 IMPLEMENTATION ARCHITECTURES**

There are many ways in which a bundle node can be instantiated. The following are some examples:

* A single process running on a general-purpose computer;
* A thread running as a background process;
* An object in an object-oriented operating system;
* A special-purpose hardware device.

NOTE – No specific instantiation is defined or expected; these decisions are purely an implementation issue.

## **2.3 SERVICES PROVIDED BY BP**

BP provides a data transmission service to move “bundles” (contiguous groups of octets) of data from one BP node to another:

1. Commencing a registration (registering a node in an endpoint);
2. Terminating a registration;
3. Switching a registration between Active and Passive states;
4. Transmitting a bundle to an identified bundle endpoint;
5. Canceling a transmission that has been requested;
6. Polling a registration that is in the Passive state;
7. Delivering a received bundle;
8. Reporting bundle status.

NOTE –Custody transfer has been removed from the Bundle Protocol specification and is expected to be realized via additional mechanisms. Services enhanced include items d) and f) of the previous list.

## **2.4 QUALITIES OF SERVICE NOT PROVIDED BY BP**

The Bundle Protocol as specified in this document does not provide the following services:

1. In-order delivery of bundles;
2. Complete delivery of sequences of bundles;
3. Insert additional capabilities and details to enable compatibility with the overall infrastructure.

These services may be provided by a layer above BP yet below the end-system applications. These services can exist as shims. Such a shim provides the logic to accomplish the desired functions and is inserted between BP and the Application Layer. This would leave the existing network protocol stack intact. This layer is described in annex E of this document.

## **2.5 SIGNIFICANT CHANGES FROM CCSDS 734.2-B-1**

Listed below are points where this specification significantly differs from CCSDS 734.2-B-1 [2]. The significant changes between RFC 5050 and RFC 9171 can be located in Appendix A of RFC 9171.

* Clarify the difference between transmission and forwarding.
* Introduce the concept of "node ID" as functionally distinct from endpoint ID, while having the same syntax.
* Restructure primary block, making it immutable. Add optional CRC.
* Add optional CRCs to non-primary blocks.
* Add block ID number to canonical block format (to support BPsec).
* Add definition of bundle age extension block.
* Add definition of previous node extension block.
* Add definition of hop count extension block.
* Remove Quality of Service markings.
* Change from SDNVs to CBOR representation.
* Add lifetime overrides.
* Time values are denominated in milliseconds, not seconds.

**2.5.1 Difference between BPv6 and BPv7 Block Types**

The current Bundle Block Types registry in the Bundle Protocol Namespace is augmented by adding a column identifying the version of the Bundle protocol (Bundle Protocol Version) that applies to the new values. IANA is requested to add the following values, as described in section 4.3.1, to the Bundle Block Types registry. The current values in the Bundle Block Types registry should have the Bundle Protocol Version set to the value "6", as shown below.

Table 2.5.1: Bundle Block Types and their associated versions

|  |  |  |
| --- | --- | --- |
| **Bundle Protocol Version** | **Description** | **CCSDS Reference** |
| None | Reserved | CCSDS 734.2-B-1 |
| 6,7 | Bundle Payload Block | CCSDS 734.2.B-1 |
| 6 | Bundle Authentication Block | CCSDS 734.2.B-1 |
| 6 | Payload Integrity Block | CCSDS 734.2.B-1 |
| 6 | Payload Confidentiality Block | CCSDS 734.2.B-1 |
| 6 | Previous-Hop Insertion Block | CCSDS 734.2.B-1 |
| 6 | ACS Payload Block | CCSDS 734.2.B-1 |
| 7 | Previous node (proximate sender) |  |
| 7 | Bundle age (in milliseconds) |  |
| 7 | Hop count (#prior xmit attempts) |  |
| 7 | Unassigned |  |
| 6,7 | Reserved for Private and/or Experimental Use | CCSDS 734.2.B-1, |

# **3. CCSDS PROFILE OF RFC 9171**

## **3.1 GENERAL**

This document adopts the Bundle Protocol as specified in Internet RFC 9171 (reference [1]), with the constraints and exceptions specified in section 3 of this document.

## **3.3 IPN NAMING SCHEME**

**3.3.1** Implementations shall support the ‘IPN’ naming scheme as referenced in section 4.2.5.1.2 of RFC 9171, *Bundle Protocol Version 7* (reference [1]).

NOTE – This standard exclusively supports the IPN naming scheme, unlike RFC 9171.

NOTE – The scheme-specific part of an IPN name consists of:

1. a sequence of ASCII numeric digits representing an integer in the range 1 to 264−1, termed the ‘node number’ of the URI (Uniform Resource Identifier);
2. an ASCII period (‘.’) character;
3. a sequence of ASCII numeric digits representing an integer in the range 0 to 264−1, termed the ‘service number’ of the URI.

NOTE - The "ipn" scheme supports the identification of BP endpoints by pairs of unsigned integers, for compact representation in bundle blocks. It is specified as follows:

Scheme syntax: This specification uses the Augmented Backus-Naur Form (ABNF) notation of [RFC5234], including the core ABNF syntax rule for DIGIT defined by that specification.

ipn-uri = "ipn:" ipn-hier-part

ipn-hier-part = node-nbr nbr-delim service-nbr ; a path-rootless

node-nbr = 1\*DIGIT

nbr-delim = "."

service-nbr = 1\*DIGIT

**3.3.2** The IPN node numbers used shall be assigned by SANA from the CCSDS CBHE Node Number Registry.

**3.3.3** The service Numbers used shall be assigned by IANA / SANA from either the IANA CBHE Service Numbers registry or the SANA CBHE Service Numbers Registry.

NOTES

1. CBHE is the compression mechanism enabled by the IPN naming scheme.
2. The SANA CBHE Node Number registry is a portion of the IANA registry that has been delegated to SANA for management by CCSDS.

## **3.5 USE OF TIME IN SECTION 4.2.6 OF RFC 9171**

A DTN time is an unsigned integer indicating the number of milliseconds that have elapsed since the DTN Epoch, 2000-01-01 00:00:00 +0000 (UTC). DTN time is not affected by leap seconds.

Each DTN time SHALL be represented as a CBOR unsigned integer item. Implementers need to be aware that DTN time values conveyed in CBOR representation in bundles will nearly always exceed (2\*\*32 - 1); the manner in which a DTN time value is represented in memory is an implementation matter. The DTN time value zero indicates that the time is unknown.

NOTE – As DTN implementations expand beyond the Earth-Moon system, longer light-times between interplanetary bodies and the Earth will make time biases difficult to manage if not properly handled. The Time WG Green Book addresses these specific concerns.

* This issue should be explored with the SEA-Time working group for handling the time biases between different interplanetary bodies and spaces.
* Within the DTN organization scheme, this tracking of accumulated time affects the Bundle Age Extension block.

# **4 SERVICE DESCRIPTION**

**4.1 SERVICES AT THE USER INTERFACE**

**4.1.1** The services provided by the Bundle Protocol shall be made available to bundle protocol users and include the following:

a) initiate a registration (registering a node in an endpoint);

b) terminate a registration;

c) switch a registration between Active and Passive states;

d) transmit a bundle to an identified bundle endpoint;

e) cancel a transmission;

f) poll a registration that is in the Passive state;

g) deliver a received bundle.

**4.1.2** The BP node shall be implemented such that virtually any number of transactions may be conducted concurrently in various stages of transmission or reception at a single BP node.

NOTE – To clarify: the implementation needs to be able to accept a primitive, and thereupon initiate a new transaction prior to the completion of previously initiated transactions. The requirement for concurrent transaction support therefore does not necessarily imply that the implementation needs to be able to begin initial transmission of data for one transaction while initial transmission of file data for one or more other transactions is still in progress. (But neither is support for this functional model precluded.)

**4.2 SUMMARY OF PRIMITIVES**

**4.2.1** The BP service shall consume the following request primitives:

– Register.request;

– Deregister.request;

– ChangeRegistrationState.request;

– Send.request;

– Cancel.request;

– Poll.request.

**4.2.2** The BP service shall deliver the following indication primitives:

– LocalBundleID.indication;

– BundleDelivery.indication.

**4.3 SUMMARY OF PARAMETERS**

**4.3.1 DESTINATION COMMUNICATIONS ENDPOINT ID**

The destination communications endpoint ID parameter shall identify the communications endpoint to which the bundle is to be sent.

NOTE – One can think of a DTN communications endpoint as an application, but in general the definition is meant to be broader. For example, a single BPA (with a single endpoint ID) could service other local nodes such as elements of a sensor network using private protocols.

**4.3.2 SOURCE COMMUNICATIONS ENDPOINT ID**

The source communications endpoint ID parameter shall uniquely identify the communications endpoint from which the bundle was sent.

**4.3.3 REPORT-TO COMMUNICATIONS ENDPOINT ID**

The report-to communications endpoint ID parameter shall identify the communications endpoint to which any bundle status reports pertaining to the bundle are sent.

**4.3.4 ISSINGLETONEID**

The IsSingletonEID parameter shall be ‘True’ if the referenced Endpoint IDentifier (EID) is a singleton, i.e., if there is at most one BP node that is a member of the endpoint identified.

**4.3.5 CLASS-OF-SERVICE PARAMETER**

**4.3.5.1** The class-of-service parameter shall indicate which class of standard procedures is to be followed when transmitting and delivering the bundle.

**4.3.5.2** The value of the class-of-service parameter shall be one of the following:

– bulk;

– normal;

– expedited.

**4.3.6 DELIVERY OPTIONS PARAMETER**

**4.3.6.1** The delivery options parameter shall indicate what optional procedures are additionally to be followed when transmitting and delivering the bundle.

**4.3.6.2** The value of the delivery options parameter shall be a combination of zero or more of the following:

a) bundle is a fragment;

b) application data unit is an administrative record;

c) bundle must not be fragmented;

d) ~~custody transfer is requested;~~

e) destination endpoint is a singleton;

f) acknowledgement by application is requested;

g) class of service;

h) request reporting of bundle reception;

i) request reporting of custody acceptance;

j) request reporting of bundle forwarding;

k) request reporting of bundle delivery;

l) request reporting of bundle deletion;

m) extended class of service.

**4.3.7 LIFETIME PARAMETER**

The lifetime parameter shall indicate the length of time, following initial creation time of a bundle, after which bundle protocol agents may discard the bundle.

**4.3.8 APPLICATION DATA UNIT PARAMETER**

The application data unit parameter shall indicate the location (in memory or non-volatile storage, a local implementation matter) of the application data conveyed by the bundle.

**4.3.9 LOCAL BUNDLE ID**

The Local Bundle ID parameter shall identify a particular bundle within the context of a given bundle protocol agent.

NOTE – This identification is provided to the user of the bundle service on submitting a bundle for transmission so that the user may later reference that bundle in other requests, such as cancellation. The form of this identifier is entirely implementation-specific and should not be confused with the Source EID and Creation Timestamp combination (global Bundle ID) used to uniquely identify bundles in the network.

**4.3.10 DELIVERY FAILURE ACTION**

**4.3.10.1** The Delivery Failure Action parameter shall identify the response the node is to take on receipt of a bundle that is deliverable subject to the registration when the registration is in the Passive state (see 4.3.11).

**4.3.10.2** The Delivery Failure Action parameter shall signal one of the following possible responses:

– defer delivery of the bundle;

– abandon delivery of the bundle.

NOTE - RFC 5050 section 3.1 contains more on when deferred bundles may be delivered to receiving applications.

**4.3.11 REGISTRATION STATE**

The Registration State is the state machine characterization of a given node’s membership in a given endpoint. A registration state must at any time be in one of two states: Active or Passive.

NOTE – A registration always has an associated ‘delivery failure action’ which denotes the action to be taken upon receipt of a bundle that is deliverable subject to the registration when the registration is in the Passive state (refer to 4.3.10). Further definition of Registration can be found in section 3.1 of RFC 5050.

**4.3.12 HEADER INFORMATION**

The Header Information parameter shall uniquely identify the delivered bundle and indicate the delivered bundle’s remaining time to live and the time of delivery to the application agent.

**4.4 BP SERVICE PRIMITIVES**

**4.4.1 Register.request**

**4.4.1.1 Function**

The Register.request primitive shall be used to notify the BP agent of the node’s membership in a communications endpoint.

**4.4.1.2 Semantics**

Register.request shall provide parameters as follows:

Register.request (delivery failure action, destination communications endpoint ID)

**4.4.1.3 When Generated**

Register.request may be generated by any BP application at any time.

**4.4.1.4 Effect on Receipt**

**4.4.1.4.1** Receipt of Register.request shall cause the BP agent to declare the node’s registration in the indicated endpoint.

NOTE – The registration is initially in Passive state.

**4.4.1.4.2** The indicated failure action shall be taken upon arrival of any bundle destined for this endpoint, as long as the registration remains in Passive state.

**4.4.1.5 Discussion—Additional Comments**

None.

**4.4.2 Deregister.request**

**4.4.2.1 Function**

The Deregister.request primitive shall be used to notify the BP agent of the end of the node’s membership in the indicated endpoint.

**4.4.2.2 Semantics**

Deregister.request shall provide parameters as follows:

Deregister.request (destination communications endpoint ID)

**4.4.2.3 When Generated**

Deregister.request may be generated by any BP application at any time when the node is registered in the indicated endpoint.

**4.4.2.4 Effect on Receipt**

Receipt of Deregister.request shall cause the node’s registration in the indicated endpoint to be rescinded.

**4.4.2.5 Discussion—Additional Comments**

Multiple nodes can be members of the same endpoint. One node deregistering from the endpoint does not affect other nodes’ delivery or delivery failure behavior.

**4.4.3 ChangeRegistrationState.request**

**4.4.3.1 Function**

The ChangeRegistrationState.request primitive shall be used to notify the BP agent of a desired change in the registration state.

**4.4.3.2 Semantics**

ChangeRegistrationState.request shall provide parameters as follows:

ChangeRegistrationState.request (destination communications endpoint ID, registrationState)

**4.4.3.3 When Generated**

ChangeRegistrationState.request may be generated by any BP application at any time when the node is registered in the indicated endpoint.

**4.4.3.4 Effect on Receipt**

**4.4.3.4.1** Receipt of ChangeRegistrationState.request shall cause the BP agent to change the state of the registration to the requested state.

**4.4.3.4.2** If the new state is Active, receipt of this request shall additionally cause the bundle protocol agent to deliver to the application all bundles, destined for the indicated endpoint, for which delivery was deferred.

**4.4.3.5 Discussion—Additional Comments**

None.

**4.4.4 Send.request**

**4.4.4.1 Function**

The Send.request primitive shall be used by the application to request transmission of an application data unit from the source communications endpoint to a destination communications endpoint.

**4.4.4.2 Semantics**

Send.request shall provide parameters as follows:

Send.request (source communications endpoint ID,

destination communications endpoint ID,

report-to communications endpoint ID,

class-of-service,

~~IsSingletonEID,~~

delivery options,

lifetime,

application data unit)

**4.4.4.3 When Generated**

Send.request may be generated by the source BP application at any time.

**4.4.4.4 Effect on Receipt**

Receipt of Send.request shall cause the BP agent to initiate bundle transmission procedures.

**4.4.4.5 Discussion—Additional Comments**

None.

**4.4.5 Cancel.request**

**4.4.5.1 Function**

The Cancel.request primitive shall be used by the application to request termination of transmission of an application data unit for which the application previously requested transmission.

**4.4.5.2 Semantics**

Cancel.request shall provide parameters as follows:

Cancel.request (Local Bundle ID)

**4.4.5.3 When Generated**

Cancel.request may be generated by the application at any time after requesting transmission of a bundle.

**4.4.5.4 Effect on Receipt**

Receipt of Cancel.request shall cause the BP agent to stop attempting to transmit and to discard the target bundle, if possible.

**4.4.5.5 Discussion—Additional Comments**

If the bundle has already been transmitted, there is no obligation on the sending BP agent to take any further action. It is an implementation matter whether a bundle that is in the process of being transmitted when a Cancel.request is received is terminated.

**4.4.6 Poll.request**

**4.4.6.1 Function**

The Poll.request primitive shall be used by the application to request immediate delivery of the least-recently received bundle that is currently deliverable subject to the node’s registration in the indicated endpoint.

**4.4.6.2 Semantics**

Poll.request shall provide parameters as follows:

Poll.request (destination communications endpoint ID)

**4.4.6.3 When Generated**

Poll.request may be generated by any BP application at any time when the node is registered in the indicated endpoint and that registration is in Passive state.

**4.4.6.4 Effect on Receipt**

Receipt of Poll.request shall cause the BP agent to deliver to the BP application the least-recently received bundle, destined for the destination communications endpoint ID, for which delivery was deferred.

NOTE – Prioritization applies only to forwarding of a bundle. Deferred bundles are delivered in the order in which they were received.

**4.4.6.5 Discussion—Additional Comments**

None.

**4.4.7 LocalBundleID.indication**

**4.4.7.1 Function**

The LocalBundleID.indication primitive shall be used to provide the application a reference to a particular bundle of which the application requested transmission.

**4.4.7.2 Semantics**

LocalBundleID.indication shall provide parameters as follows:

LocalBundleID.indication (Local Bundle ID)

**4.4.7.3 When Generated**

LocalBundleID.indication shall be generated by a BP agent once it has consumed a Send.request from the application.

**4.4.7.4 Effect on Receipt**

The effect on receipt of LocalBundleID.indication by a BP application is undefined.

**4.4.7.5 Discussion—Additional Comments**

On receiving this notice the sending application can, for example, release resources of its own that are allocated to the bundles being transmitted, or remember the Local Bundle ID so that transmission can be canceled in the future if necessary.

**4.4.8 BundleDelivery.indication**

**4.4.8.1 Function**

The BundleDelivery.indication primitive shall be used to indicate to the bundle service user that a bundle has been delivered to the application.

**4.4.8.2 Semantics**

BundleDelivery.indication shall provide parameters as follows:

BundleDelivery.indication (header information,

application data unit)

**4.4.8.3 When Generated**

BundleDelivery.indication shall be generated by a BP agent upon delivery of a bundle, either on reception of bundles destined for active registrations or in response to poll requests referencing passive registrations.

**4.4.8.4 Effect on Receipt**

The effect on receipt is defined by the application.

**4.4.8.5 Discussion—Additional Comments**

None.

# **5 SERVICES BP REQUIRES OF THE SYSTEM**

## **5.1 RELIABLE STORAGE REQUIREMENTS**

BP nodes shall have access to a reliable storage service.

NOTES

1. This storage mechanism may be in dynamic memory or via a persistent mechanism such as a solid-state recorder and may be organized by various means to include file systems.
2. The implementation of this storage can be shared among multiple elements of the communication stack so that reliability mechanisms at multiple layers do not have to maintain multiple copies of the data being transmitted.
3. Volume of storage required and duration of storage are mission- and implementation-dependent.

## **5.2 UNDERLYING COMMUNICATION SERVICE REQUIREMENTS**

**5.2.1** The following information shall be available to BP, either from the local operating environment or from the underlying communication service provider:

* forward advancing time that can be represented as ‘DTN time’ as defined by RFC 9171 (reference [1]);
* ~~at least one long-lived singleton EID of which the node is a member;~~
* a unique creation time for BP traffic.

NOTE – The means by which this information is accessed by BP is implementation-dependent.

**5.2.2** Each convergence layer protocol adapter is expected to provide the following services to the bundle protocol agent:

* sending a bundle to a bundle node that is reachable via the convergence layer protocol;
* notifying the bundle protocol agent of the disposition of its data sending procedures with regard to a bundle, upon concluding those procedures;
* delivering to the bundle protocol agent a bundle that was sent by a bundle node via the convergence layer protocol.

NOTE - The convergence layer service interface specified here is neither exhaustive nor exclusive. That is, supplementary DTN protocol specifications (including, but not restricted to, the Bundle Protocol Security [BPSEC]) may expect convergence layer adapters that serve BP implementations conforming to those protocols to provide additional services such as reporting on the transmission and/or reception progress of individual bundles (at completion and/or incrementally), retransmitting data that were lost in transit, discarding bundle-conveying data units that the convergence layer protocol determines are corrupt or inauthentic, or reporting on the integrity and/or authenticity of delivered bundles.

In addition, bundle protocol relies on the capabilities of protocols at the convergence layer to minimize congestion in the store-carry-forward overlay network. The potentially long round-trip times characterizing delay-tolerant networks are incompatible with end-to-end reactive congestion control mechanisms, so convergence-layer protocols MUST provide rate limiting or congestion control.

NOTE – BP nodes have the capability to receive identification of the previous hop from that hop. It is strongly advised, but optional, that connecting entities to/of the DTN implementation use this capability to identify themselves when sending data.

**5.2.3** The service provided by the protocols beneath BP (not necessarily by the convergence layer protocol itself) shall deliver only complete layer-(N−1) service data units (bundles) to the receiving BP Node.

**5.2.4** The service provided by the underlying protocols (not necessarily by the convergence layer protocol itself) shall provide integrity checking of the layer-(N−1) service data units (bundles) and shall discard layer-(N−1) service data units that are determined to be corrupted.

**5.2.5** The convergence layer adaptor service may provide a cap on the rate at which a sending BP engine can inject data into the layer-(N−1) service.

**5.2.6** Delivery of duplicate BP PDUs to a BPA by the underlying layer shall be acceptable.

# **6 CONFORMANCE REQUIREMENTS**

**6.1 General Requirements**

**6.1.1 Protocol Implementation**

A conforming implementation of this protocol shall:

* conform to the BP-bis specification (RFC 9171, reference [1]);
* ~~conform to the CBHE specification (RFC 6260, reference [5]);~~
* implement the modifications in section 3 of this document;
* implement the services described in section 4 of this document.

**6.1.2 PICS PROFORMA**

An implementer shall prepare a Protocol Implementation Conformance Statement (PICS) based on the defined proforma in annex A of this document.

## **6.2 BUNDLE PROTOCOL REQUIREMENTS**

**6.2.1 MAJOR CAPABILITIES**

**6.2.1.1 All Bundle Protocol Implementations**

All BP implementations for CCSDS shall implement and/or conform to the following:

1. bundle structure as described in RFC 9171 sections 3.1, 4.0, 4.1, 4.2.3, 4.2.5.1, 5.8, and 9;
2. block structure as described in RFC 9171 sections 4.1, 4.2.1, 4.2.1 4.3, 4.3.1, 4.3.2, , 4.4, 4.4.1, 4.4.2, 4.4.3;
3. administrative record generation and structure as described in RFC 9171 section 5.1, 6.0, 6.1, and 6.2;
4. administrative record processing as described in RFC 9171 sections 6.1.1~~, 6.1.2, and 6.3~~;
5. CBHE in accordance with RFC 6260 and section 3 of this document:
6. BP nodes shall use CBHE endpoint identifiers obtained from SANA;
7. BP nodes shall use CBHE service numbers obtained from SANA or, in the case of mission-private services, may use service numbers from the range reserved for private/experimental use;

**6.2.1.2 Bundle Protocol Senders**

**6.2.1.2.1** A conforming BP implementation shall support the following in accordance with the base standard:

a) bundle transmission as described in RFC 9171 sections 3.3, 4.2.4, 5.12, and 5.2;

b) bundle forwarding as defined in RFC 9171 sections 4.2.3, 5.1, 5.3, 5.4, 5.4.1, 5.4.2, and 5.5).

**6.2.1.2.2** In addition, a BP sender shall also support the following capabilities in accordance with the base standard:

a) intermittent connectivity conditions specified in RFC 9171 section 1;

b) late binding as described in RFC 9171 section 1;

c) bundle delivery failure as defined in RFC 9171 section 3.1;

d) bundle priority as defined in RFC 9171 section 4.2.3 ~~and the ECOS specification of annex C~~;

e) bundle deletion procedures as defined in RFC 9171 sections 3.1, 4.2.3, 5.10, and 5.14;

f) dictionary byte array and revision per RFC 9171 sections 4.2.5.1 ~~and 4.7~~.

**6.2.1.3 Bundle Protocol Receivers**

A conforming BP implementation shall support the following in accordance with the base standard:

1. bundle acceptance in accordance with RFC 9171 sections 4.2, 4.3.1, 4.3.2, 5.6, 5.7, 5.9, and 5.10~~, and 5.13~~;
2. node registration as defined in RFC 9171 sections 3.3 ~~and 5.16~~.

# **ANNEX A** **Protocol Implementation Conformance Statement Proforma (Normative)**

**Overview**

This annex provides the Protocol Implementation Conformance Statement (PICS) Requirements List (RL) for CCSDS-compliant implementations of BP. The PICS for an implementation is generated by completing the RL in accordance with the instructions below. An implementation shall satisfy the mandatory conformance requirements of the base standards referenced in the RL.

An implementation’s completed RL is called the PICS. The PICS states which capabilities and options of the protocol have been implemented. The following can use the PICS:

1. the protocol implementer, as a checklist to reduce the risk of failure to conform to the standard through oversight;
2. the supplier and acquirer or potential acquirer of the implementation, as a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the standard PICS proforma;
3. the user or potential user of the implementation, as a basis for initially checking the possibility of interworking with another implementation (it should be noted that, while interworking can never be guaranteed, failure to interwork can often be predicted from incompatible PICSes);
4. a protocol tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

**Instructions for Completing the RL**

An implementer shows the extent of compliance to the protocol by completing the RL; that is, compliance to all mandatory requirements and the options that are not supported are shown. The resulting completed RL is called a PICS. In the Support column, each response shall be selected either from the indicated set of responses, or it shall comprise one or more parameter values as requested. If a conditional requirement is inapplicable, N/A should be used. If a mandatory requirement is not satisfied, exception information must be supplied by entering a reference Xi, where i is a unique identifier, to an accompanying rationale for the noncompliance.

## **A3 Notation**

**A3.1** The symbols in table A‑1 are used in the RL to indicate the status of features.

**Table** **A‑1: PICS Notation**

|  |  |
| --- | --- |
| **Symbol** | **Meaning** |
| M | Mandatory |
| O | Optional |
| O.<n> | Optional, but support of at least one of the group of options labeled by the same numeral <n> is required |

**A3.2** The symbols in table A‑2 shall be used in the ‘Support’ column of the PICS.

**Table** **A‑2: Symbols for PICS ‘Support’ Column**

|  |  |
| --- | --- |
| **Symbol** | **Meaning** |
| Y | Yes, the feature is supported by the implementation. |
| N | No, the feature is not supported by the implementation. |
| N/A | The item is not applicable. |

## **A4 Referenced Base Standards**

**A4.1** The base standards referenced in the RL shall be:

1. CCSDS BP (this document);
2. RFC 9171 (reference [1]);
3. ~~RFC 6260 (reference [5]).~~

**A4.2** In the tables below, the notation in the Reference column combines one of the short-form document identifiers above (e.g., RFC 5050) with applicable subsection numbers in the referenced document. RFC numbers are used to facilitate reference to subsections within the Internet specifications.

A5 General Information

**Identification of PICS**

|  |  |  |
| --- | --- | --- |
| **Ref** | **Question** | **Response** |
| 1 | Date of Statement (DD/MM/YYYY) |  |
| 2 | PICS serial number |  |
| 3 | System conformance statement cross-reference |  |

**Identification of Implementation Under Test (IUT)**

|  |  |  |
| --- | --- | --- |
| **Ref** | **Question** | **Response** |
| 1 | Implementation name |  |
| 2 | Implementation version |  |
| 3 | Name of hardware (machine) used in test |  |
| 4 | Version of hardware (machine) used in test |  |
| 5 | Name of operating system used during test |  |
| 6 | Version of operating system used during test |  |
| 7 | Additional configuration information pertinent to the test |  |
| 8 | Other information |  |

**Identification**

|  |  |  |
| --- | --- | --- |
| **Ref** | **Question** | **Response** |
| 1 | Supplier |  |
| 2 | Point of contact for queries |  |
| 3 | Implementation name(s) and version(s) |  |
| 4 | Other information necessary for full identification (e.g., name(s) and version(s) for machines and/or operating systems |  |

**Protocol Summary**

|  |  |  |
| --- | --- | --- |
| **Ref** | **Question** | **Response** |
| 1 | Protocol version |  |
| 2 | Addenda implemented |  |
| 3 | Amendments implemented |  |
| 4 | Have any exceptions been required?  NOTE – A YES answer means that the implementation does not conform to the protocol. Non-supported mandatory capabilities are to be identified in the PICS, with an explanation of why the implementation is non-conforming. | 1. Yes 2. No |
| 5 | Date of statement (DD/MM/YYYY) |  |

**Basic Requirements**

| **Item** | **Protocol Feature** | **Reference** | **Status** | **Support** |
| --- | --- | --- | --- | --- |
| IPN\_naming | Use of ‘IPN’: EID Naming Scheme | RFC 6260 section 2.1 | M |  |
| CBHENodeNo | Use IPN node numbers assigned by SANA | This document: 3.2.2 | M |  |
| CBHE Service No | Use CBHE service numbers assigned by IANA / SANA | This document: 3.2.3 | M |  |
| CBHE Encoding of EIDs by CLAs | Encoding of BP endpoints via CBHE | RFC 6260 section 2.2; this document section B2.2 | O |  |
| bpECOS | Extended Class of Service Block as defined in reference [4] | This document: 3.3, C2, C3 | O |  |
| bpTime | Precision of onboard time | RFC 5050 section 6.1; this document: 3.4 | M |  |
| LTP ‘raw’ CL adaptor |  | This document: B3.1.2.1 | O.1 |  |
| LTP SDA CL adaptor |  | This document: B3.1.2.1 | O.1 |  |
| UDP CL adaptor |  | This document: B4 | O.1 |  |
| bpInitRegistration | BP registration initialization service | This document: 4.1 | M |  |
| bpTerm Registration | BP terminate initialization service | This document: 4.1 | M |  |
| bpSwitchRegistration | BP switch registration state service | This document: 4.1 | M |  |
| bpTransmitBundle | BP transmit a bundle to an identified bundle endpoint service | This document: 4.1 | M |  |
| bpCancelTransmission | BP cancel a transmission service | This document: 4.1 | M |  |
| bpPollRegistration | Poll a registration that is in Passive state | This document: 4.1 | M |  |
| bpDeliverRecvdBundle | Deliver received bundle service | This document: 4.1 | M |  |
| bpBundle | Bundle structure | RFC 5050: sections 3.1, 4.0, 4.2, 4.4, and 5.8.8 | M |  |
| bpBlock | Primary Bundle Block structure | RFC 5050: sections 4.1, 4.5, 4.5.1, 4.5.2 | M |  |
| bpBlockPayload | Payload Bundle Block Structure | RFC 5050: sections 3.1, 4.5, 4.5.3, and 4.7 | M |  |
| SDNV | Self-Delimiting Numeric Values | RFC 5050: section 4.1 | M |  |
| SDNVLargeValue | SDNV encoded value larger than 264−1 | RFC 5050: section 4.1 | O |  |
| bpPriority | Bundle Priority | RFC 5050: section 4.2; this document: annex C | M |  |
| bpAccept | Bundle Acceptance | RFC 5050: sections 4.2, 4.5.1, 4.5.2, 5.6, 5.7, 5.9, 5.10, 5.13 | M |  |
| bpTransmission | Bundle Transmission Service | RFC 5050: sections 3.3, 4.3, 5.15, 5.2 | M |  |
| bpForward | Bundle Forwarding | RFC 5050: sections 4.2, 5.1, 5.3, 5.4, 5.4.1, 5.4.2, 5.5 | M |  |
| bpDelete | Bundle deletion procedures | RFC 5050: sections 3.1, 4.2, 5.13, and 5.14 | M |  |
| DictionaryArray | Dictionary byte array | RFC 5050: sections 4.4 and 4.7 | M |  |
| bpExtnBlk | Implementation supports extension blocks | RFC 5050: section 4.6 | M |  |
| Admin Record Structure | Administrative Record Definition | RFC 5050: sections 5.1, 6.0, 6.1, and 6.2 | M |  |
| Admin Record Processing | Administrative Record Processing | RFC 5050: sections 6.1.1, 6.1.2, and 6.3 | M |  |
| Admin Record Generation | Administrative Record Generation | RFC 5050: sections 5.1 and 6.2 | M |  |
| Connectivity | Intermittent connectivity conditions | RFC 5050: section 1 | M |  |
| LateBinding | Late Binding | RFC 5050: section 1 | M |  |
| bpDeliveryFail | Bundle delivery failure | RFC 5050: section 1 | M |  |
| CustodySigProc | Custody Signal Processing | This document: 6.2.1.3b) | M |  |
|  | Custody countdown timer |  | O |  |
|  | Custody transfer failure action |  | O |  |
| EIDRegisterNode | Registration of node in an endpoint | RFC 5050: section 3.3 | M |  |
| EIDTerminateNode | Termination of node in an endpoint | RFC 5050: section 3.3. | M |  |
| EIDSwitchNode | Switching node registration between Active & Passive | RFC 5050: section 3.3 | M |  |
| EIDPollNode | Polling node registration | RFC 5050: sections 3.3 and 5.16 | M |  |
| EIDFormat | EIDs may be expressed in some internationalized manner (IRI) | RFC 5050: section 4.4 | O |  |
| BlkFwdError | Blk forward w/o processing flag may be optionally cleared by another node that receives the bundle and can process that block | RFC 5050: section 4.6 | O |  |
| Dictionary revision | Modification of the BP dictionary to support changes in custodian | RFC 5050 section 4.7; this document: 6.2.1.2.2 | O |  |
| Fragmentation | Splitting a bundle into multiple fragments (each of which is a bundle in its own right) | RFC 5050 section 5.8 | O |  |
| Reassembly | Reassembly of bundle fragments into the original bundle (performed at the destination) | RFC 5050 section 5.9 | M |  |
| MIB\_state | Bundle State Information | This document: table F‑1 | M |  |
| MIB\_errors | Error and Reporting Information | This document: table F‑2 | M |  |
| MIB\_registration | Registration Information | This document: table F‑3 | M |  |
| MIB\_CL\_info | Convergence-Layer Information | This document: table F‑4 | M |  |
| MIB\_Config | General Configuration Information | This document: annex F | M |  |

# **ANNEX B** **Convergence LAyer aDapters (Normative)**

## **B1 Overview**

This annex describes various Convergence Layer Adapters (CLAs) to support mission operations both in space and on the ground. There are many possible convergence layer protocols to support the various communications interfaces with which the Bundle Protocol may interact. This annex is in no manner comprehensive or rigorous but contains CCSDS supported CLAs that have been demonstrated under various environments, have been requested to be included at the time of this writing, and appear applicable to CCSDS users.

When a specific CLA appears to contradict the specification for that specific convergence layer or the Bundle Protocol specification, the CLA will be assumed to be invalid.

## **B2 Convergence Layer Adapters**

**B2.1 Available CL ADAPTERs**

Compliant implementations shall implement at least one of the CLAs in this section.

**B2.2 Compressed Bundle Header Encoding**

Convergence Layer Adapters shall support the compressed bundle header encoding mechanisms of RFC 6260 [5].

**B2.3 Layering Covergence Layers**In order to utilize DTN protocols across the Internet, whether for testing purposes or as part of a larger network path, it is highly recommended to encapsulate them into a standard Internet Protocol so that they travel easily across the Internet. This is particularly true for LTP, which provides no endpoint addressing. This encapsulation choice needs to be made carefully in order to avoid redundancy, since DTN protocols may provide their own reliability mechanisms. See RFC 7122 for more details.

B3 TCP convergence Layer adapter

When sending/receiving bundles using UDP at the convergence layer, bundles shall be encapsulated in TCP packets according to the Delay-Tolerant Networking TCP Convergence-Layer Protocol [RFC 9174].

B4 UDP convergence Layer adapter – Encapsulation of bundles in udP datagrams

When sending/receiving bundles using UDP at the convergence layer, bundles shall be encapsulated in UDP datagrams as follows:

b) each bundle shall be encapsulated into one UDP datagram with no additional bytes;

NOTE – Bundle protocol agents should endeavor to bound the upper limit of transmitted bundles to the underlying network MTU (minus overheads). If the bundle to be encapsulated is larger than the maximum UDP MTU size, the bundle needs to be fragmented at the bundle layer before transmission.

c) all implementations should use UDP port 4556/UDP;

d) all implementations should ensure that the traffic sent by the UDP convergence layer adaptor does not adversely affect other traffic on the network;

NOTE –Network characteristics can best be managed on a closed network or a network with reserved bandwidth; or the utilization of congestion control procedures as described in RFC 5405 (reference [8]) can be adopted.

NOTEbundle protocol agents should endeavor to send bundles of such a size as not to require fragmentation by the IP (Internet Protocol) layer.

NOTE – In practice this generally means keeping the size of the IP datagram (including the IP and UDP headers, plus the bundle) to less than 1500 bytes. When using UDP as the convergence layer protocol, bundles are limited to a maximum size of 65,535 bytes (including all of the bundle blocks, the 8-byte UDP header, and the IP header (20 bytes for IPv4, 40 bytes for IPv6).

B5 LTP convergence Layer adapter

Convergence Layer Adapters shall support the compressed bundle header encoding mechanisms of RFC 6260 [5].

B5.1 Encapsulation of Bundles in LTP Blocks

B5.1.1 General

When sending/receiving bundles using LTP (reference [11]) at the convergence layer, bundles shall be encapsulated in LTP blocks as described in the following subsections.

B5.1.2 RELIABLE TRANSMISSION VIA LTP

**B5.1.2.1** For reliable bundle transmission, bundles shall be encapsulated in LTP blocks containing only red-part (reliable) data.

**B5.1.2.1.1** Bundles shall be encapsulated either

1. as a single bundle per LTP block with no leading or trailing bytes: in this case the Destination LTP Client Service ID shall be the service ID for ‘Bundle Protocol’ as specified in the SANA LTP Client Service ID Number Registry (reference [7]);
2. according to the Client Operations section (section 7) of the LTP-for-CCSDS Book (reference [9]): the Destination LTP Client Service ID provided by the LTP CLA to the LTP service shall be the service ID for ‘Bundle Protocol’ as specified in the SANA LTP Client Service ID Number Registry (reference [7]).

NOTE – In this case the LTP SDA (Service Data Aggregation) service will use Client Service ID 2 (Service Data Aggregation) as the client service ID for the LTP block; the transmitted LTP block will contain the LTP Client Service ID for ‘Bundle Protocol’ as the first bytes of the payload.

B5.1.3 UNRELIABLE TRANSMISSION VIA LTP

For unreliable bundle transmission, bundles shall be encapsulated into LTP blocks containing only green-part (unreliable) data. In this case one bundle shall be encapsulated in each LTP block with no leading or trailing bytes. The LTP Client Service ID shall be the service ID for ‘Bundle Protocol’ as specified in the SANA LTP Client Service ID Number Registry (reference [7]).

## **B6 CCSDS ENCAPSULATION SERVICE CONVERGENCE LAYER ADAPTER—ENCAPSULATION OF BUNDLES VIA THE CCSDS ENCAPSULATION SERVICE**

When sending/receiving bundles using the CCSDS Encapsulation Service (reference [10]) at the convergence layer, bundles shall be encapsulated via the Encapsulation Service specified in the SANA Protocol Identifier for Encapsulation Service Registry (reference [7]) as follows:

1. each bundle shall be presented as the data unit of one invocation of one ENCAPSULATION.request function of the Encapsulation Service with no additional leading or trailing bytes;
2. the Data Unit Loss Flag (if present at the receiver) may be used by the receiving CLA in an implementation-specific manner.

## **B7 USLP CONVERGENCE LAYER ADAPTER**

When sending/receiving bundles using the Unified Space Link Protocol (USLP) at the convergence layer, bundles shall be encapsulated in USLP packets as follows:

Should be like AOS, but with more options. Degenerative case, looks like AOS. General case, offers:

Big or small bundles allowance (variable frames)

Alerts (High priority)

Security for different bundles.

## **B8 SPACE PACKETS**

# **Annex C** **AGGREGATE CUSTODY SIGNAL Specification (Normative)**

## **C1 OVERVIEW**

**C1.1 GENERAL**

This annex defines an administrative bundle type and a complementary bundle block that carries extra information and enhances custody transfer of BP as introduced in RFC 5050 and adapted for RFC BIBECT. Enhanced custody transfer is particularly important with asymmetric routing and data rates between forward (uplink) and return (downlink) paths in many environments including human and robotic spaceflight.

**C1.2 INTRODUCTION**

In order to guarantee delivery of data, BP provides for the capability to positively identify a bundle and acknowledge the receipt of that bundle. This capability is provided for each bundle discretely. For links with asymmetric data rates, the acknowledgement on a bundle-by-bundle basis with link asymmetries of two or three orders of magnitude may be onerous. By the aggregation of custody signals, link efficiencies of more than one order of magnitude can be realized.

An Aggregate Custody Signal (ACS) is similar to a normal custody signal in that it signals acceptance or rejection of custody and a reason for this acceptance or rejection. An ACS extends the BP custody mechanism by identifying one or more bundles in a compressed format.

The identification is in the form of blocks or fills, like TCP selective acknowledgments. The ACS block is a contiguous sequence of custody IDs that identify specific bundles. These custody IDs are provided in each bundle’s Custody Transfer Enhancement Block (CTEB). An aggregate custody signal is the payload of a bundle with the ‘Administrative Record’ flag set. As a payload, it is contained inside a payload block. The aggregate custody signal is a new Administrative Record, number 4.

## **C2 DEFINITION OF TERMS**

A pending ACS is a logical entity that shall contain the following:

a) custodian EID;

b) reason code;

c) ‘successful’ flag;

d) ACS generation countdown timer;

e) bundle IDs.

An Aggregate Custody Signal is an administrative record that shall have:

a) an administrative record type 4 for ‘Aggregate Custody Signal’;

b) Administrative Record Flag ‘record is for a fragment’ cleared.

## **C3 ACS COMPONENT FORMATS**

**C3.1 OVERVIEW**

An ACS function is similar to a normal custody signal in signaling acceptance or rejection of custody and a reason for this acceptance or rejection. An ACS provides additional benefits by providing custody signals for one or more bundles in a compressed format.

The identification is in the form of blocks or fills, like TCP selective acknowledgments. A block is a contiguous sequence of custody IDs that identify bundles. The custody IDs come from Custody Transfer Enhancement Blocks.

All block formats and behavior are consistent with RFC 5050 and RFC BIBECT.

**C3.2 AGGREGATE CUSTODY SIGNAL**

**C3.2.1** An aggregate custody signal shall be the payload of a bundle with the ‘Administrative Record’ flag set.

NOTE – As a payload, it is contained inside a payload block.

**C3.2.2** The aggregate custody signal shall be the Administrative Record, number 4.

|  |  |  |
| --- | --- | --- |
| 0x04 | Status |  |
| Left edge of first fill\* | | Length of first fill\* |
| Difference between right edge of first fill and left edge of second fill\* | | Length of second fill\* |
| • • • | | |
| Difference between right edge first N-1 and left edge of fill N\* | | Length of fill N\* |

\* Field is a Concise Binary Object Representation (CBOR)

**Figure C-1: ACS Payload Block Definition**

**C3.2.3** The first field shall identify administrative record type 4.

**C3.2.4** The second field shall be a ‘Status’ byte encoded in the same way as the status byte for administrative records in RFC 9171, using the same reason codes.

**C3.2.5** The third field shall be the custody ID of the left edge of the first fill, encoded as a CBOR.

**C3.2.6** The fourth field shall be the length of the first fill, encoded as an CBOR, indicating the number of contiguous custody IDs described by this fill.

NOTE – All fields beyond this are optional: if there is only one fill in this ACS, then the payload block ends. After this, fills continue with a difference field and a length field.

**C3.2.7** The next field shall be the difference between the right edge of the first block and the left edge of the second block. It is encoded as an CBOR.

**C3.2.8** The next field shall be the length of the second block, encoded as an CBOR.

**C3.3 CUSTODY TRANSFER ENHANCEMENT BLOCK**

**C3.3.1** A CTEB is required for each bundle that is to be supported by ACS and is the responsibility of the accepting bundle protocol agent that supports ACS processing.

**C3.3.2** For bundle protocol agents in the network which are not ACS aware, the block flags must be set to ensure that the block is passed through the network unimpeded.

|  |  |  |
| --- | --- | --- |
| 0x0a | Block Flags\* | Block Length |
| Custody ID\* | | CTEB Creator Custodian EID† |
| \*Field is a CBOR | | †Field is variable length |

**Figure C-2: CTEB Block Definition**

**C3.3.3** There shall be only one CTEB per bundle.

**C3.3.4** The first field shall identify block type 0x0a.

**C3.3.5** The second field shall be block flags encoded as a CBOR.

**C3.3.6** The third field shall be the block length encoded as an CBOR.

**C3.3.7** The fourth field shall be a non-negative integer, an identifier encoded as an CBOR, which uniquely identifies a bundle for this custodian at this instant in time.

**C3.3.8** The fifth field shall be the custodian ID of the creator of the CTEB.

## **C4 ACS BEHAVIOR**

**C4.1 DISCUSSION**

Custody signals are a verbose form of bundle acknowledgement. Whereas a minimum custody signal is composed of 43 bytes, of which 20 bytes are the primary bundle block, 22 bytes are custody signal for a single bundle, and a single byte can represent a scheme specific node, as defined by RFC 5050 and RFC BIBECT.

The use of ACS allows for a high degree of compression because of two factors: Custody ID from a CTEB, and the exclusion of the overhead of a primary bundle block for each custody signal. Time is omitted since it is not used for processing but is logged and can be approximated by the ACS bundle creation time. Therefore an ACS bundle single acknowledgement is at most 13 bytes:

– 1 byte for the Administrative Record Type and Flags per RFC 9171;

– 1 byte for status per RFC 9171;

– 10 bytes for the left edge of the first fill assuming that 262−1 is the largest Custody ID;

– 1 byte for the length of the first fill of a single aggregated bundle.

Additional bundles aggregated will increase the compression by only adding the aggregated bundle custody signal without the overhead of the primary bundle block.

Fragmentation of a bundle creates additional bundles of a smaller size with proportionate fragment offsets and fragment lengths as identified in RFC 5050 and RFC BIBECT. Therefore a fragmented ACS is consistent with a fragmented bundle where the payload is the ACS ‘Failed’ or ‘Succeeded’ list fragment.

**C4.2 ACS IMPLEMENTATION**

**C4.2.1 Discussion**

There are four possibilities when a bundle protocol agent accepts a bundle with a CTEB.

a) For an intermediate node which is not ACS capable and accepts custody, the bundle protocol agent ignores the CTEB and updates the custodian field in the primary bundle block. Since the CTEB custodian is not updated, the CTEB is invalid, and the next ACS-capable bundle protocol agent will delete the CTEB.

b) For an intermediate node which is not ACS capable and does not accept custody, the bundle protocol agent forwards the bundle without change. The CTEB is not recognized.

c) For an intermediate node which is ACS capable and does not accept custody, possibility b) is the mode of operation. A node may not accept a bundle for any of a number of reasons as defined in RFC 5050.

d) For an intermediate node which is ACS capable and accepts custody, the bundle protocol agent compares the CTEB custodian with the primary bundle block custodian. If they are different, the CTEB is invalid and deleted. For identical custodians, the primary bundle block and CTEB are updated with the new custodian by the bundle protocol agent, and custody aggregation is utilized to improve link efficiency.

Item d) bounds the defining set of capabilities unique to ACS. By accepting a bundle for custody transfer, an ACS-capable bundle protocol agent will process the bundle per RFC 5050, section 5.10.1. However, instead of the normal custody signaling, the CTEB identifies in shortened form the specific bundle by custody ID. Figure D-3 features a detailed ACS processing flow of the following requirements.

**C4.2.2** Requirements

**C4.2.2.1** Non-ACS-aware bundle protocol agents shall process ACS-supporting bundles per RFC 9171 section 5.10.

**C4.2.2.2** For ACS-aware bundle protocol agents which do not accept custody of ACS:

a) for bundles without a valid CTEB block as identified in RFC 5050 section 5.10, the bundle protocol agent shall generate a ‘Failed’ status;

b) for bundles with a valid CTEB:

1) the bundle protocol agent shall aggregate ‘Failed’ status into a single bundle as identified in D3.2:

i) the aggregation of ‘Failed’ status shall not exceed the maximum allowed bundle size;

ii) the time period for aggregation of bundle status shall not exceed the maximum allowed;

2) the bundle protocol agent shall transmit an ACS as identified in RFC 5050 section 5.10;

3) the bundle protocol agent shall delete, upon successful transmission of an ACS signal, the associated timer and pending ACS ‘Failed’.

**C4.2.2.3** For ACS-aware bundle protocol agents which do accept custody of ACS:

a) the bundle protocol agent shall generate a ‘Succeeded’ status for bundles without a valid CTEB block as identified in RFC 5050 section 5.10;

b) the bundle protocol agent shall update the custodian of the Primary Bundle Block and the CTEB as identified in D3.3;

c) for bundles with a valid CTEB:

1) the bundle protocol agent shall aggregate ‘Succeeded’ status into a single bundle as identified in D3.2:

i) the aggregation of ‘Succeeded’ status shall not exceed maximum allowed bundle size;

ii) the time period for aggregation of bundle status shall not exceed the maximum allowed;

2) the bundle protocol agent shall delete, upon successful transmission of an ACS signal, the associated timer and pending ACS ‘Succeeded’.

**C4.2.2.4** A non-ACS-aware bundle protocol agent shall forward unchanged an ACS to the originating Custody EID.

**C4.2.2.5** An ACS-aware bundle protocol agent that receives an ACS shall retrieve each bundle ID associated with each Custody ID.

**C4.2.2.6** An ACS-aware bundle protocol agent that receives an ACS shall execute RFC 5050 section 6.3 for each bundle ID of the ACS signal.

**C4.2.2.7** An ACS-aware bundle protocol agent shall utilize the ACS bundle timestamp time as the ‘Time of Signal’ when executing RFC 5050 section 6.3.



**Figure C-3: ACS Processing Flow**

# **ANNEX E**

**BP MANAGED INFORMATION**

**(NORMATIVE)**

## **E1 BASIC REQUIREMENTS**

**E1.1** Each BP node shall support a set of managed information that represents the state of the node at a particular time. The minimal set of such information contains those data items identified by RFC 5050 and collected in this annex. This collection of managed information is shown as the MIB on the right of figure 1-1.

**E1.2** BP nodes shall support five types of managed information:

a) bundle state information;

b) error and reporting information;

c) registration information;

d) convergence layer information;

e) node state information.

**E1.3** In addition to required information, each BP node may choose to provide supplementary information. Each identified managed information item shall identify whether its collection and accurate reporting is required or recommended.

NOTES

1 Representation of, and mechanisms for access to, managed information items will be implementation matters.

2 Individual pieces of managed information may describe related events. Care must be taken when modifying these data to ensure that related data sets remain coherent. For example, when a cumulative counter ‘rolls over’ or is otherwise reset, related counters should also be reset.

**E2** **BUNDLE STATE INFORMATION**

**E2.1** **OVERVIEW**

Bundles do not have a natural end state within a node: they are forwarded, delivered, or deleted. As such, bundles at rest within a node exist pending a particular action. This set of managed information describes these bundle states and the transitions between them.

**E2.2 Supported Types of Bundle State Information**

BP nodes shall support the bundle state information itemized in table E‑1.

**Table** **E‑1: Bundle State Information**

|  |  |  |  |
| --- | --- | --- | --- |
| **Managed Information Item** | **Description** | | **Req?** |
| **Retention Constraints** | | | |
| Bundles Retained for Forwarding | The number of bundles/bytes associated with the retention constraint ***forward pending*** at this node. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| Bundles Retained for Transmission | The number of bundles/bytes associated with the retention constraint ***dispatch pending*** at this node. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| Bundles Retained for Custody Acceptance | The number of bundles/bytes associated with the retention constraint ***custody accepted*** at this node. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| Bundles Retained for Reassembly | The number of bundles/bytes associated with the retention constraint ***reassembly pending*** at this node. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| **Priority Counters** | | | |
| Bulk Bundles Sourced | The number of bundles/bytes generated by this node with the ***bulk*** priority. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| Normal Bundles Sourced | The number of bundles/bytes generated by this node with the ***normal*** priority. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| Expedited Bundles Sourced | The number of bundles/bytes generated by this node with the ***expedited*** priority. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| Bulk Bundles Queued | The number of bundles/bytes with the ***bulk*** priority currently resident on this node. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| Normal Bundles Queued | The number of bundles/bytes with the ***normal*** priority currently resident on this node. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| Expedited Bundles Queued | The number of bundles/bytes with the ***expedited*** priority currently resident on this node. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| **Fragmentation** | | | |
| Fragmentation | The number of bundles that have been fragmented by this node. | Cumulative Bundles | Yes |
| Number of Fragments | The number of fragments created by this bundle. | Cumulative Bundles | Yes |

**E3 NODE ERROR AND REPORTING INFORMATION**

**E3.1 Overview**

Nodes generate reports in response to both anomalous and special events. This set of managed information reports on the number of errors and reports constructed at the node.

**E3.2 Supported Types of Error and Reporting Information**

BP nodes shall support the error and reporting information itemized in table E‑2.

**Table** **E‑2: Error and Reporting Information**

| **Managed Information Item** | **Description** | | **Req?** |
| --- | --- | --- | --- |
| **Bundle Deletions** | | | |
| No Info Deletions | The number of bundles deleted with the ***No additional information*** reason code. | Cumulative Bundles | No |
| Expired Deletions | The number of bundles deleted with the ***Lifetime expired*** reason code. | Cumulative Bundles | No |
| Uni-forwarded Deletions | The number of bundles deleted with the ***Forwarded over unidirectional link*** reason code. | Cumulative Bundles | No |
| Cancellation Deletions | The number of bundles deleted with the ***Transmission canceled*** reason code. | Cumulative Bundles | No |
| No Storage Deletions | The number of bundles deleted with the ***Depleted Storage*** reason code. | Cumulative Bundles | No |
| Bad EID Deletions | The number of bundles deleted with the ***Destination endpoint ID unintelligible*** reason code. | Cumulative Bundles | No |
| No Route Deletions | The number of bundles deleted with the ***No known route to destination from here*** reason code. | Cumulative Bundles | No |
| No Timely Contact Deletions | The number of bundles deleted with the ***No timely contact with next node on route*** reason code. | Cumulative Bundles | No |
| Bad Block Deletions | The number of bundles deleted with the ***Block unintelligible*** reason code. | Cumulative Bundles | No |
| Bytes deleted | The total number of bytes in all bundles deleted at this node. | Cumulative Bytes | No |
| **Bundle Processing Errors** | | | |
| Failed Custody Transfers | The number of incoming bundles/bytes whose request for custody was not successful at this node. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| Failed Forwards | The number of bundles/bytes that have experienced a forwarding failure at this node. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| Abandoned Delivery | The number of bundles/bytes whose delivery has been abandoned at this node. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |
| Discarded Bundles | The number of bundles/bytes discarded at this node. | Cumulative Bytes | No |
| Cumulative Bundles | Yes |

**E4 REGISTRATION INFORMATION**

**E4.1 Overview**

Each node registers in one or more endpoints. These registrations allow for the reception and processing of bundles in the context of the endpoints to which they are addressed.

**E4.2 Supported Types of Registration Information**

BP nodes shall support the registration information itemized in table F‑3.

**Table** **E‑3: Registration Information**

| **Managed Information Item** | **Description** | **Req?** |
| --- | --- | --- |
| **Identity Information** | | |
| Endpoint Identifier | The Endpoint ID of this registered endpoint. | Yes |
| Activity State | The current state of the EID, at the time the managed information was queried.  One of: ACTIVE or PASSIVE. | Yes |
| Singleton State | Whether this EID is a singleton EID.  One of: YES or NO. | Yes |
| Default Failure Action | The default action to be taken when delivery is not possible.  One of: ABANDON or DEFER. | Yes |

**E5 node state INFORMATION**

**E5.1 Overview**

Global node state information provides the context for using other managed information items.

**E5.2 Supported Types of node state Information**

BP nodes shall support the node state information itemized in table F‑4.

**Table** **E‑4: Node State Information**

| **Managed Information Item** | **Description** | **Req?** |
| --- | --- | --- |
| **Identity Information** | | |
| Node Identifier | The Endpoint ID that uniquely and permanently identifies this node. | Yes |
| Bundle Protocol version number | The number of the version of the Bundle Protocol that is supported at this node. | Yes |
| Available storage | The number of kilobytes of storage allocated to bundle retention at this node and not currently occupied by bundles. | Yes |
| Last up time | The most recent time at which the operation of this node was started or restarted. | Yes |
| Registration count | The number of different endpoints in which this node has been registered since it was last started or restarted. | No |
| **Extension Information (one occurrence per extension)** | | |
| Extension name | The name identifying one of the BP extensions supported at this node. | Yes |

# **ANNEX F**

**SECURITY, SANA, AND PATENT CONSIDERATIONS**

**(INFORMATIVE)**

**F1 SECURITY**

**F1.1 OVERVIEW**

The Bundle Protocol as defined by RFC 9171 has factored in security from the outset of its design. The necessary security architecture and services have been developed in an accompanying RFC, the Bundle Protocol Security specification. Because BP was designed for a resource-constrained environment, it is essential to ensure that only those entities authorized to utilize those resources be allowed to do so.

Also, because of the long latencies and delays in the constrained environments which utilize BP, integrity and confidentiality are essential. Without adequate protections in place to ensure that data integrity and confidentiality are maintained, the difficulty in identifying compromised data will be compounded as a result of the unique environment of CCSDS missions.

**F1.2 SECURITY CONCERNS WITH RESPECT TO THE CCSDS DOCUMENT**

The BP specification (reference [1]) contains a security section (9), which addresses necessary measures to protect bundle protocol data and recommends the use of the Bundle Security Protocol (BPSec) of RFC BPSec. Two types of security blocks are defined in RFC BPSec:

1. Bundle Integrity Block (BIB) - Used to ensure the integrity of its plain text security target(s). The integrity information in the BIB MAY be verified by any node along the bundle path from the BIB security source to the bundle destination. Waypoints add or remove BIBs from bundles in accordance with their security policy. BIBs are never used for integrity protection of the cipher text provided by a BCB. Because security policy at BPSec nodes may differ regarding integrity verification, BIBs do not guarantee hop-by-hop authentication, as discussed in RFC BPsec Section 1.1.
2. Block Confidentiality Block (BCB) - Indicates that the security target(s) have been encrypted at the BCB security source in order to protect their content while in transit. The BCB is decrypted by security acceptor nodes in the network, up to and including the bundle destination, as a matter of security policy. BCBs additionally provide integrity protection mechanisms for the cipher text they generate.

**F1.3 AUDITING OF RESOURCE USAGE**

No mechanisms are defined in this specification to audit or assist with the auditing of resource usage by the protocol.

**F1.4 POTENTIAL THREATS AND ATTACK SCENARIOS**

No potential threat or attacks scenarios are discussed.

**F1.5 CONSEQUENCES OF NOT APPLYING SECURITY TO THE TECHNOLOGY**

By not applying the native security of the BP protocol and the extended security of BPSec allowed by BP, the system must rely on security measures provided at the CLA interfaces. For space applications these may be non-existent or merely physical because of the lack of integration between payload and ground systems interfaces. If no security is applied at the BP or lower layers, then applications may be open to man-in-the middle attacks, replay attacks, or a general loss of integrity of transported bundles.

**F2 SANA CONSIDERATIONS**

The recommendations of this document request SANA to create the following registries based on the recommendation of 3.6:

a) the registry named Bundle Protocol Compressed Bundle Header Encoding Node Numbers which consists of a table of parameters:

1) the initial registry values are not defined;

2) the registration rule for new values of this registry requires no engineering review, but the request must come from the official representative of a space agency, member of the CCSDS;

b) the registry named Bundle Protocol Compressed Bundle Header Encoding Service Numbers which consists of a table of parameters:

1) limited initial registry values are defined;

2) the registration rule for new values of this registry requires an official representative of a space agency member of the CCSDS request followed by expert CESG review.

**F3 PATENT CONSIDERATIONS**

There are no known patents covering the Bundle Protocol as described in this document and its normative references.

# **ANNEX G**

## **SANA Registry Considerations**

**G1 IPN NODE Numbers**

**G1.1 General**

SANA has established the registry

<https://sanaregistry.org/r/bp_cbhe_node_numbers/bp_cbhe_node_number.html>

to manage CBHE NODE Number assignments. The registry shall be used to catalog agency-managed BP CBHE Node Numbers and LTP engine IDs that are coincident

NOTE – The purpose of this registry is to ensure uniqueness of BP CBHE Node Numbers used in space missions

**G1.2 SANA BP CBHE Node Number Registration Policy**

The registration policy for the registry shall be: no engineering review required; request must come from an identified CCSDS representative of a member, observer, or affiliate organization.

**G2 CBHE SERVICE NUMBERS**

**G2.1 General**

SANA has established the registry

<https://sanaregistry.org/r/bp_cbhe_service_numbers/bp_cbhe_service_number.html>

to manage CBHE Service Number assignments. The registry shall be used by CCSDS to catalog BP CBHE Service Numbers that denote different bundle services.

NOTE – The purpose of this registry is to ensure uniqueness of the CBHE Service Numbers used in space missions

**G2.2 SANA BP CBHE Service Number Registration Policy**

The registration policy for the registry shall be: no engineering review required; request must come from an identified CCSDS representative of a member, observer, or affiliate organization.

# ANNEX H

MIN MAX

Will be filling this annex with the minimum maximum from Simon’s BPv7 Bundle Data Elements workbook in next update.

# ANNEX I

CSR

# **ANNEX J**

**INFORMATIVE REFERENCES**

**(INFORMATIVE)**

[H1] *Rationale, Scenarios, and Requirements for DTN in Space.* Issue 1. Report Concerning Space Data System Standards (Green Book), CCSDS 734.0-G-1. Washington, D.C.: CCSDS, August 2010.

[H2] J.H. Saltzer, D.P. Reed, and D.D. Clark. “End-to-End Arguments in System Design.” In *Proceedings of the 2nd International Conference on Distributed Computing Systems (April 8-10, 1981, Paris, France),* 509-512. Los Alamitos, CA, USA: IEEE Computer Society, 1981.

[H3] *Organization and Processes for the Consultative Committee for Space Data Systems.* Issue 4. CCSDS Record (Yellow Book), CCSDS A02.1-Y-4. Washington, D.C.: CCSDS, April 2014.

# **ANNEX K**

**ABBREVIATIONS AND ACRONYMS**

**(INFORMATIVE)**

Term Meaning

AA application agent

ACS aggregate custody signal

ACS-EXB aggregate custody signal external block

BIB bundle integration block

BCB bundle confidentiality block

BP Bundle Protocol

BPA bundle protocol agent

BPSec Bundle Protocol Security

CBHE compressed bundle header encoding

CBOR Concise Binary Object Representation

CCSDS Consultative Committee for Space Data Systems

CRC cyclic redundancy check

CTEB custody transfer enhancement block

CL convergence layer

CLA convergence layer adapter

DTN delay tolerant network

EID endpoint identifier

IANA Internet Assigned Numbers Authority

IP Internet Protocol

IPN Interplanetary Network

LTP Licklider Transmission Protocol

OSI Open Systems Interconnection

PICS protocol implementation conformance statement

PDU payload data unit

RL requirements list

RFC Request For Comment

SANA Space Assigned Numbers Authority

SDA service data aggregation

SDU service data unit

TCP Transmission Control Protocol

UDP User Datagram Protocol

URI uniform resource identifier