The Consultative Committee for Space Data Systems

Research and Development for
Space Data System Standards

CCSDS BUNDLE PROTOCOL SPECIFICATION

EXPERIMENTAL SPECIFICATION

CCSDS 734.20-O-1

ORANGE BOOK
April 2025

The Consultative Committee for Space Data Systems

Research and Development for Space Data System Standards

CCSDS BUNDLE PROTOCOL SPECIFICATION

EXPERIMENTAL SPECIFICATION

CCSDS 734.20-O-1

ORANGE BOOK

April 2025

AUTHORITY

Issue:	Experimental Specification, Issue 1
Date:	April 2025
Location:	Not Applicable

This document has been approved for publication by the Consultative Committee for Space Data Systems (CCSDS). 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).

This document is published and maintained by:

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

FOREWORD

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. 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 document 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 addressed to the CCSDS Secretariat at the address indicated on page i.

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

Member Agencies

- Agenzia Spaziale Italiana (ASI)/Italy.
- Canadian Space Agency (CSA)/Canada.
- Centre National d'Études 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 Science Policy Office (BELSPO)/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.
- China 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.
- Hellenic Space Agency (HSA)/Greece.
- Indian Space Research Organization (ISRO)/India.
- Institute of Space Research (IKI)/Russian Federation.
- Korea Aerospace Research Institute (KARI)/Korea.
- Ministry of Communications (MOC)/Israel.
- Mohammed Bin Rashid Space Centre (MBRSC)/United Arab Emirates.
- National Institute of Information and Communications Technology (NICT)/Japan.
- National Oceanic and Atmospheric Administration (NOAA)/USA.
- National Space Agency of the Republic of Kazakhstan (NSARK)/Kazakhstan.
- National Space Organization (NSPO)/Chinese Taipei.
- Naval Center for Space Technology (NCST)/USA.
- Netherlands Space Office (NSO)/The Netherlands.
- Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- Scientific and Technological Research Council of Turkey (TUBITAK)/Turkey.
- South African National Space Agency (SANSA)/Republic of South Africa.
- Space and Upper Atmosphere Research Commission (SUPARCO)/Pakistan.
- Swedish Space Corporation (SSC)/Sweden.
- Swiss Space Office (SSO)/Switzerland.
- United States Geological Survey (USGS)/USA.

PREFACE

This document is a CCSDS Experimental Specification. Its Experimental status indicates that it is part of a research or development effort based on prospective requirements, and as such it is not considered a Standards Track document. Experimental Specifications are intended to demonstrate technical feasibility in anticipation of a 'hard' requirement that has not yet emerged. Experimental work may be rapidly transferred onto the Standards Track should a hard requirement emerge in the future.

DOCUMENT CONTROL

Document	Title and Issue	Date	Status
CCSDS 734.20-O-1	CCSDS Bundle Protocol Specification, Experimental Specification, Issue 1	April 2025	Original Issue

CONTENTS

Figure

Table

1 INTRODUCTION

1.1 PURPOSE

The purpose of this document is to establish a CCSDS Experimental Specification for Bundle Protocol (BP), based on the bundle protocol of Request for Comments (RFC) 9171 (reference [1]), which defines the end-to-end protocol, bundle structure, naming schemes, and block types for the exchange of messages (bundles) that support Delay/Disruption Tolerant Networking (DTN). This document includes abstract service descriptions for the application services provided by BP. This document does not describe how to route bundles in a DTN. It also does not address how BP can be used to provide data reliability and/or accountability.

1.2 SCOPE

This Experimental Specification is designed to be applicable to any space mission or space mission network infrastructure that might benefit from delay and/or disruption tolerance. It is intended that this Experimental Specification become a uniform standard among all CCSDS Agencies.

This Experimental Specification is intended to be applicable to all systems that claim conformance to the CCSDS BP version 7.

In addition to telemetry and telecommand, BP can function over a wide range of CCSDS and Internet or ground-based protocols, such as Advanced Orbiting Systems (AOS), Encapsulation Packet Protocol (EPP), Licklider Transmission Protocol (LTP), Proximity-1 Space Link Protocol, Space Packet Protocol (SPP), Transmission Control Protocol (TCP), Unified Space Link Protocol (USLP), and User Datagram Protocol (UDP).

The CCSDS believes it is important to document the rationale underlying the recommendations chosen so that future evaluations of proposed changes or improvements will not lose sight of previous decisions. The concept and rationale for the use of the BP in space links may be found in reference [G1].

1.3 ORGANIZATION OF THE EXPERIMENTAL SPECIFICATION

This Experimental Specification is organized as follows:

- Section 2 contains an overview of the BP 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 from the hosting system.
- Section 6 contains conformance requirements.

– Annex A contains the Protocol Implementation Conformance Statement (PICS) proforma.

– Annex B contains the Convergence Layer Adapters (CLAs).

– Annex C contains BP Managed Information.

– Annex D contains Security, Space Assigned Numbers Authority (SANA), and Patent considerations.

– Annex E contains BP Element Nomenclature.

– Annex F contains the Interplanetary Internet (ipn) Uniform Resource Identifier (URI) Scheme Updates.

– Annex G contains Informative References.

– Annex H contains descriptions of implementations that have successfully tested this specification.

– Annex I contains Abbreviations and Acronyms used in this document.

1.4 DEFINITIONS

1.4.1 DEFINITIONS FROM OPEN SYSTEMS INTERCONNECTION SERVICE DEFINITION CONVENTIONS

This Experimental Specification makes use of several terms defined in reference [2]. As used in this Recommended Standard, those terms are to be interpreted in a generic sense, that is, 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 OPEN SYSTEMS INTERCONNECTION BASIC REFERENCE MODEL

This Experimental Specification makes use of several terms defined in reference [3]. As used in this Experimental Specification, those terms are to be understood in a generic sense, that is, 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.

1.4.3 DEFINITIONS FROM RFC 9171

1.4.3.1 Overview

This Experimental Specification makes use of several 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 one or more 'convergence layer adapters', and an 'application agent'. The major components are illustrated in figure 1-1 ('CLx PDUs' are the PDUs of the convergence-layer protocols used in individual networks).

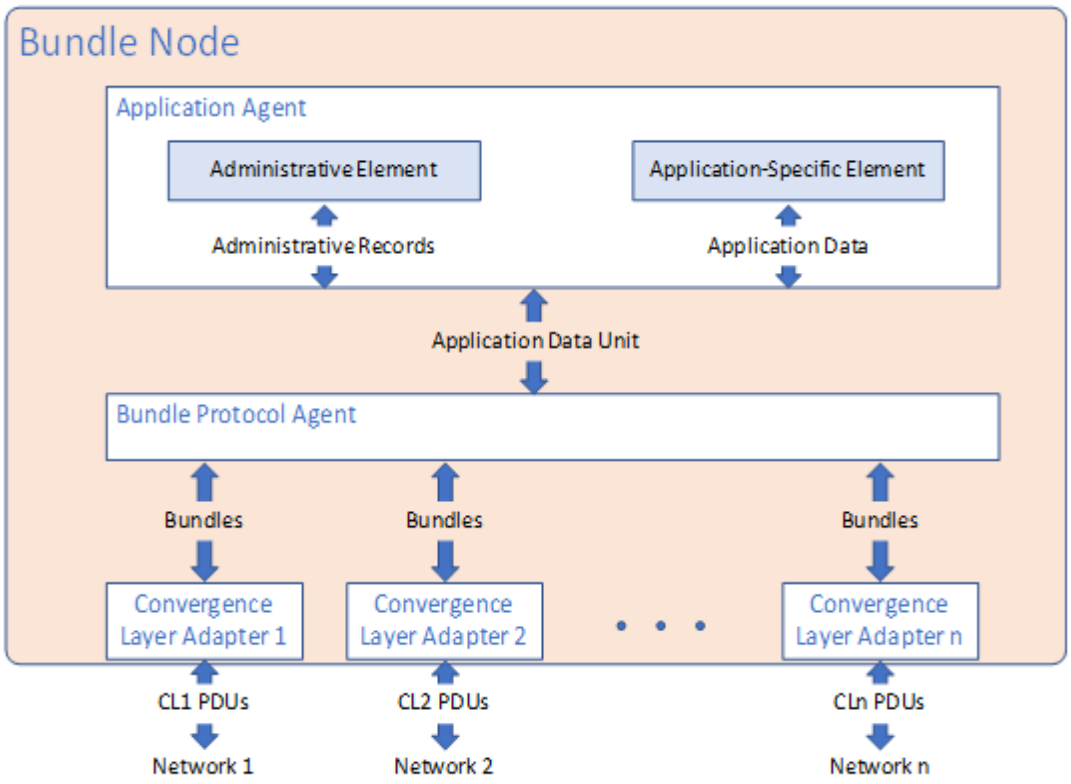

Figure 1-1: Graphical Representation of a Bundle Node

There is one application agent (AA) per conceptual bundle node. It may provide communication services to multiple applications, and the node may register in multiple

endpoints (or may provide multiple endpoint identifiers to the bundle protocol agent [BPA], requesting delivery of bundles to any of those endpoints).

NOTE – Even Bundle Nodes that perform solely DTN Routing/Forwarding functions must still implement an AA to provide the Administrative Element. The presence of an Application Specification Element depends on the implementation use case.

1.4.3.2 RFC 9171-Derived Terms

administrative element, AE: In the context of an application agent, the node component that constructs and requests transmission of administrative records (defined in 6.1 of RFC9171), including status reports, and accepts delivery of and processes any administrative records that the node receives.

application agent, AA: A 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.

application data unit, ADU: The application-specific data being transferred via the BP. The data in an ADU is carried in the payload block of a bundle and may be split among the payloads of multiple bundles if the original bundle is fragmented.

application-specific element, ASE: In the context of an application agent, the node component that constructs, requests transmission of, accepts delivery of, and processes units of user application data.

block: One of the BP data structures that together constitute a well-formed bundle.

bundle endpoint, endpoint: A set of zero or more bundle nodes that all identify themselves for BP purposes by some common identifier, called a ‘bundle endpoint ID’ (or, in this document, simply ‘endpoint identifier’); endpoint IDs are described in detail in RFC9171 Section 4.2.5.1.

bundle node, node: 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, BPA: A node component that offers the BP services and executes the BP procedures.

bundle: 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.

convergence layer adapter, CLA: A node component that sends and receives bundles on behalf of the BPA, utilizing the services of some 'integrated' protocol stack that is supported in one of the networks within which the node is functionally located.

endpoint identifier, EID: A text string identifying a bundle endpoint (see RFC 9171, section 3.1). Each Endpoint Identifier (EID) is a URI. As such, each EID can be characterized as having this general structure:

< scheme name > : < scheme-specific part, or 'SSP' >

fragment, fragmentary bundle: A bundle whose payload block contains a partial payload.

registration: The state machine characterizing a given node's membership in a given endpoint. Any single registration has an associated delivery failure action as defined in RFC 9171 and must at any time be in one of two states: Active or Passive. Registrations are local; information about a node's registrations is not expected to be available at other nodes, and the BP does not include a mechanism for distributing information about registrations. An Active registration is one in which the BPA attempts immediate delivery of bundles to applications; a Passive registration is one in which the BPA processes the bundle according to the delivery-failure action for the registration (i.e., either to store the bundle for later delivery to the application or to abandon it).

1.5 REFERENCES

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

[1] S. Burleigh, K. Fall, and E. Birrane. *Bundle Protocol Version 7.* RFC 9171. Reston, VA: ISOC, January 2022. https://datatracker.ietf.org/doc/rfc9171/

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

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

[4] B. Sipos, et al. *Delay-Tolerant Networking TCP Convergence-Layer Protocol Version 4.* RFC 9174. Reston, VA: ISOC, January 2022. https://datatracker.ietf.org/doc/rfc9174

[5] *Space Packet Protocol.* Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 133.0-B-2. Washington, D.C.: CCSDS, June 2020.

[6] *Encapsulation Packet Protocol.* Issue 3. Recommendation for Space Data System Standards (Blue Book), CCSDS 133.1-B-3. Washington, D.C.: CCSDS, May 2020.

[7] "Protocol Identifier for Encapsulation Service." Space Assigned Numbers Authority. https://sanaregistry.org/r/protocol_id.

[8] J. Postel. *User Datagram Protocol.* RFC 768. Reston, Virginia: ISOC, August 1980.

[9] *Licklider Transmission Protocol (LTP) for CCSDS.* Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 734.1-B-1. Washington, D.C.: CCSDS, May 2015.

[10] C. Bormann. *Concise Binary Object Representation (CBOR) Sequences.* RFC 8742. Reston, VA: ISOC, February 2020. https://datatracker.ietf.org/doc/html/rfc8742.

[Only references required for the implementation of the specification are listed in the References subsection. (See reference [1] for additional information on this subsection.)]

2 OVERVIEW

2.1 GENERAL

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

– Intermittent connectivity

  - Link connectivity within an interplanetary environment can periodically experience Loss of Signal (LOS) due to a variety of factors, including solar conjunction, occultation, atmospheric signal dispersion, etc.

  - Link connectivity in a near-Earth environment may periodically experience loss of signal due to obstructions, atmospheric signal dispersion, etc.

– Variable delays, which may be large and irregular

  - Delays in data transmission between nodes will occur in interplanetary (and larger) scale environments. This delay is caused mostly by the extreme distance data can be required to travel. Delay can also be caused by events like planetary occultation, in which a planetary body may inhibit signal transmission.

  - Delays may also occur in smaller scale (e.g., near-Earth) environments, for example, resulting from contention for scarce scheduled resources such as antenna transmission opportunities, power constraints on duty cycles, or transient loss of connectivity.

– Highly variable transmission error rates

  - Error characteristics may vary widely at different links along the end-to-end path and/or at different times because of external factors.

  - For near-Earth missions, error rates may be strongly affected by various factors, such as elevation angle.

– Asymmetric and simplex links

  - Deep space missions often carry constraints regarding the amount of equipment they can support on the satellite. Spacecraft telecommunication resources are generally optimized to ensure compliance with data download requirements for the prevailing instruments. The result of this resource optimization is an asymmetric, sometimes even simplex, link between the satellite and the receiver.

  - Asymmetries may also occur in near-Earth missions as a result of asymmetric hardware.

– Disparate data rates

  - Data rates may vary greatly at different links along the end-to-end path. Thus, a very high-rate link may impinge on a node with a low-rate output, requiring the node to buffer traffic for a significant period of time.

One core element of DTN is the BP. It 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. This concept is illustrated in figure 2-1, in which BP is used to provide an end-to-end data delivery service over an internetwork (on the left) and a link-layer hop (on the right). Wherever the data path transits the bundle layer in the diagram, data may be stored waiting for an outbound path to become available or for delivery to an AA.

Key BP capabilities include:

– the ability to use physical mobility to assist in the forwarding of data;

– the ability to respond to signaling from reliable CLAs to move the responsibility for retransmission from node to node;

– the ability to cope with intermittent connectivity, including cases in which the sender and receiver are not concurrently present in the network;

– the ability to take advantage of scheduled, predicted, and opportunistic connectivity, whether bidirectional or unidirectional, in addition to continuous connectivity;

– the ability to use available bandwidth for a wide variety of services and functions;

– the late binding of BP network EIDs to underlying constituent network addresses.

Reference [1] contains descriptions of these capabilities and rationale for the DTN architecture.

BP uses underlying ‘native’ Data Link Layer 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 (CL x), for example, could represent an adapter specific to a Transmission Control Protocol (TCP) network. The CLA on the right (CL y) could represent an interface to the Licklider Transmission Protocol (LTP) (reference [9]), 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, for example, an on-orbit internet and a ground internet, terrestrial or otherwise.

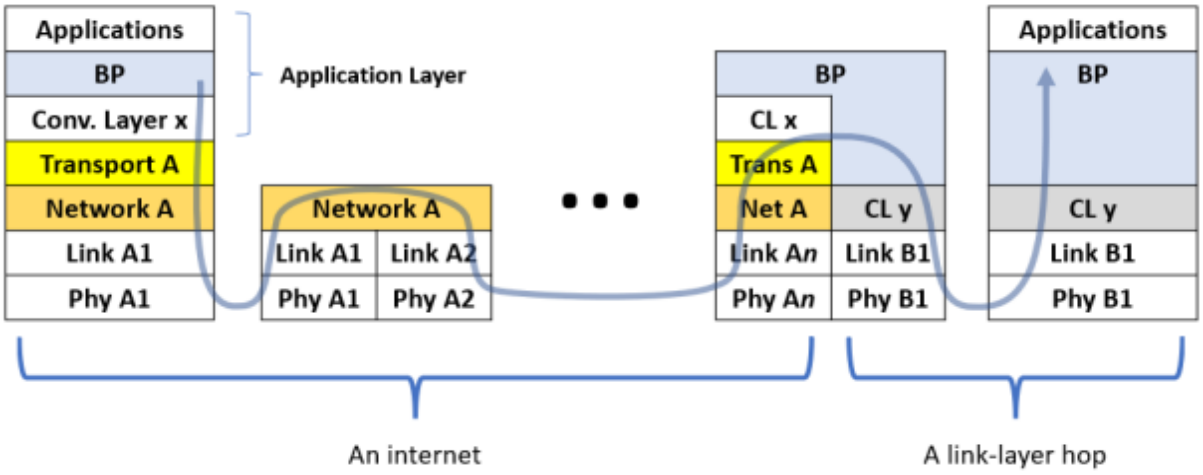

Figure 2-1: Bundle Protocol End-to-End Delivery Service

RFC 9171 describes the format of the messages (called bundles) passed between nodes participating in bundle transmission. Additionally, it addresses endpoint naming and describes how the protocol may be extended to support new capabilities while maintaining compatibility with the base protocol. Neither RFC 9171 nor this document address bundle routing algorithms (e.g., Schedule-Aware Bundle Routing [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 BP has improved the protocol in terms of simplicity and flexibility since the protocol was first released in CCSDS 734.2-B-1. These improvements make BP Version 7 (BPv7) incompatible with its previous iteration. Therefore, this document, upon publication, will obsolete CCSDS 734.2-B-1.

BP supports end-to-end communications that may include austere environments in which more commonly known communications protocols (e.g., TCP/IP) tend to break down and stop functioning. In such scenarios, the BP is an excellent technological innovation that allows multiple internetworking environments in previously unconnected locations to interact.

2.2 NODES, ENDPOINTS, AND THEIR IDENTIFIERS

RFC 9171 defines a bundle endpoint, endpoint identifier, bundle node, bundle node identifier, and bundle node number. What follows is a succinct discussion to summarize these concepts and help disambiguate between them.

A bundle endpoint is defined as a set of zero or more bundle nodes that all identify themselves for BP purposes by some common identifier, called a ‘bundle EID’. None, one, or several bundle nodes may be registered in a single common endpoint, resulting in a possibly empty multicast endpoint. Additionally, a bundle node may be registered in multiple bundle endpoints, with this configuration being more common in current DTN deployments.

Section 3.2 of RFC 9171 defines the concept of the 'null' endpoint, which is an endpoint with no members, identified by a special EID called 'null' EID. Within the ipn URI scheme, the 'null' EID is represented by the 'null' ipn URI. This means that the URIs dtn:none (Section 4.2.5.1.1 of RFC 9171), ipn:0.0, and ipn:0.0.0 all refer to the BPv7 'null' endpoint.

Bundles are by definition created by identified nodes, and they are destined for endpoints. 'Anonymous bundles' – that is, bundles for which the source node identifier is the null EID – are the only exception to this rule. They are defined in RFC 9171 to enable DTN to support applications where anonymity of the sender is important.

Given that bundle nodes and bundle endpoints are decidedly different concepts, uniquely distinguishing them requires two sets of identifiers, one for nodes and one for endpoints, which are termed 'node IDs' and 'bundle EIDs'. However, rather than defining separate namespaces for each of them, RFC 9171 instead uses EIDs for both. This choice is justified by two factors:

– First, every bundle node has an administrative agent as part of its AA, which must be able to exchange administrative records with other bundle nodes via the BPA. To enact this exchange, each bundle node must be permanently and structurally registered to a singleton endpoint known as the 'administrative endpoint'. Hence RFC 9171 requires the EID of a node's administrative endpoint may also serve as its node ID, uniquely identifying it.

– Second, because it is common practice for a bundle node to be registered in multiple singleton EIDs, RFC 9171 also allows any of these EIDs to serve as node IDs for the bundle node. Evidently, non-singleton EIDs cannot be used as node IDs.

RFC 9171 specifies that endpoint identifiers are URIs and, as such, have a general structure of the form <scheme name> : <scheme-specific part, or SSP>, where the scheme defines a set of syntactic and semantic rules to parse and interpret the SSP. In turn, this specification requires compliant implementations to adhere to the ipn URI scheme (see 3.2.1), which defines each EID as a URI in the form of 'ipn:node-nbr.service-nbr', in which, by definition, node numbers are the first part of the SSP. Furthermore, the ipn URI scheme requires all endpoints to be singletons, hence allowing them to act as node IDs. Combined, these facts allow node numbers to be used as a mnemonic and a convenient way to distinguish between nodes. However, node numbers are not, by themselves, node IDs as previously defined.

2.3 SERVICES PROVIDED BY BUNDLE PROTOCOL

BP provides a data transmission service to move 'bundles' (contiguous groups of octets) of data from one BP node to another. The specific services provided at the service interface are:

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

b) terminating a registration;

c) switching a registration between Active and Passive states;

d) transmitting a bundle to an identified bundle endpoint;

e) polling a registration that is in the Passive state;

f) delivering a received bundle;

g) report on status of bundle send request (note BundleSendRequest.indication).

2.4 QUALITIES OF SERVICE NOT PROVIDED BY BUNDLE PROTOCOL

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

a) in-order delivery of bundles;

b) assured delivery of bundles;

c) deduplication;

d) broadcast, multicast, or anycast bundle delivery.

NOTE – Provision of any of these services may be achieved using mechanisms external to this specification. For example, custody transfer is omitted from this document and may be standardized later via additional mechanisms, possibly supported by extension blocks. In the context of this specification, one way to increase probability of delivery is to use only reliable CLAs and/or an application-level reliability mechanism.

2.5 ONGOING AND FUTURE WORK

2.5.1 INTRODUCTION

This specification covers the core BP functionality and does not include specifications of security or network management.

2.5.2 SECURITY

The CCSDS DTN Working Group (WG) is currently standardizing a set of security services based on Bundle Protocol Security (BPSec) IETF RFC 9172 (reference [G3]). BPSec provides per-block (or per-group-of-blocks) security services, including cryptographic integrity and confidentiality. With the ‘base’ BPv7 protocol, there is no mechanism to prevent a node from ‘spoofing’ transmitted bundles by using the source EID of another node. While such attacks might be detectable by closely examining routing, there is no guarantee that such mechanisms would work or be sufficient.

In addition to the CCSDS BPSec Blue Book under development, the DTN WG will, together with the CCSDS Security WG, develop a Blue or Magenta Book of CCSDS security contexts and recommended policies. The intent is to recommend that implementations use BPSec to provide integrity to at least the primary block of a bundle, and preferably to the combination

of the primary and payload blocks. Even without standardized key management/key distribution, users should have the option to choose algorithms that provide the ability to cryptographically authenticate the primary block, which includes the source EID. For instance, a shared secret key between the sender and receiver would provide authentication of the sender, as would a public-private key pair that includes a certificate that allows the receiver to verify the correctness of a signature generated by the source.

The goal is to eventually provide an automated, scalable key management system. Such a system is currently prototyped in the Interplanetary Overlay Network (ION) implementation as Delay-Tolerant Key Administration (DTKA). DTKA would need to be standardized, along with capabilities on which it relies (e.g., bundle multicast), which would need to be incorporated into the BPv7 specification suite if DTKA were to be widely deployed.

2.5.3 NETWORK MANAGEMENT

There will be many configuration parameters that need to be managed for each bundle node. There is ongoing work in the Space Internetworking Services (SIS)-DTN WG and in the IETF to standardize a network management protocol that provides a level of autonomy in resource-constrained environments. The DTN management architecture (DTNMA) (reference [G4]) is the current draft specification. DTNMA is structured to provide an overall management protocol and set of encoding rules for a set of Asynchronous Data Models (ADMs). The community (both SIS-DTN and IETF) envisions a set of ADMs that includes both basic specification-level ADMs (e.g., an ADM that describes the configuration and monitoring of a 'stock' BPv7 bundle node) and implementation-specific ADMs (e.g., an ADM that includes information specific to a particular BPv7 implementation).

The benefits of standardizing a network management protocol tend to be more relevant to monitoring than they are to configuration. That is, while an agency might allow some other agency to monitor various configuration parameters of a bundle node, it seems unlikely that an agency would allow another agency to configure that node. The WG does expect to include capabilities such as control/configuration of contact plan information.

Network Management, and particularly configuration changes, may need to be secured using the BP Security protocol This would allow a node to reject configuration changes that do not pass cryptographic checks.

2.6 MECHANICS OF JOINING THE NETWORK

This subsection describes, at a high level, the mechanics of inserting a new node into an existing BPv7 network. While the network is still small, these manual procedures should suffice, although they are not expected to scale as the network grows. With network expansion, the procedures described here will likely shift to more automated, service-based solutions.

a) Node number(s) to use for the nodes need to be determined. Node numbers are managed by SANA.

b) The existing BP node(s) to which the new node will connect need to be determined.

c) The Service Site and Apertures (SS&A) SANA registry currently includes information about which sites provide DTN services. This document requests the extension of the SS&A SANA registry to include, with each site that provides DTN services, the node numbers of the DTN nodes at the site.

d) The SS&A SANA registry includes Point-Of-Contact (POC) information for sites. Site POCs can establish connectivity to the BP node(s) at the sites. Operators of new BP nodes need to confer with the site POCs or their designees to agree on convergence layers, contact plans, and connection specifics. Service sites may be at fixed locations (on Earth or other planetary bodies), or they may be hosted on spacecraft of different types.

e) Operators of new nodes need to communicate with the operators of the nodes with which they wish to communicate (as destinations) to agree on security policies and other requirements. Those endpoints may or may not implement BPSec, or they may have other implementation-specific mechanisms (e.g., firewall-like capabilities). It is expected that all nodes will eventually implement BPSec.

f) Users who wish to receive network monitoring information need to work with the individual BP node managers to determine how to receive that information. Network management is not yet standardized (by either CCSDS or IETF), so custom solutions are to be expected.

g) Connecting to a BP network means that anybody on that network can potentially send bundles to the new node. Users should consider implementing BPSec and establishing security policies to prevent unwanted traffic from being delivered to their applications.

3 CCSDS PROFILE OF RFC 9171

3.1 BUNDLE PROTOCOL FROM RFC 9171

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

3.2 NAMING SCHEMES

3.2.1 Implementations of this specification shall deliver and/or forward bundles whose source, destination, and report-to endpoint identifiers use the ipn URI naming scheme as defined in section 4.2.5.1.2 of RFC 9171, *Bundle Protocol Version 7* (reference [1]), subject to policy.

NOTES

1 Node number 0 is reserved in the ipn URI scheme.

2 Annex F provides additional information on the ipn URI scheme.

3.2.2 Implementations of this specification may forward bundles whose source, destination, or report-to endpoint identifier is the 'null' identifier.

NOTE – Bundles with a source identifier equal to the null identifier are anonymous bundles.

3.2.3 Implementations shall use ipn node numbers assigned by organizations that are documented in the SANA CCSDS Compressed Bundle Header Encoding (CBHE) Node Number Registry.

3.2.4 Implementations shall use service numbers assigned by Internet Assigned Numbers Authority (IANA)/SANA from the IANA 'ipn' Scheme URI Well-Known Service Numbers for BPv7 registry.

NOTES

1 The IANA registry includes a private address space of Service Numbers that can be used for mission-specific purposes.

2 The 'CBHE' label was adopted before BPv7 was standardized; the name was enshrined in registries and is therefore used in the name of the SANA CCSDS node number registry.

3.3 BUNDLE CREATION

3.3.1 Bundles shall be assigned source node ID and creation timestamps when Application Data Units (ADUs) are accepted for transmission by the BPA.

3.3.2 The combination of source node ID and creation timestamp shall be returned to the sending application in the bundle send request indication.

3.3.3 The source node IDs of all non-anonymous bundles sourced by a given BPA shall have the same node number.

NOTE – Users may use different service numbers in the source node IDs of bundles sent.

3.3.4 Implementations of this specification are not required to be able to source bundles with sending EID is the null identifier (anonymous bundles).

3.4 BUNDLE CANCELLATION

Implementations of this specification are not required to implement the 'Canceling a Transmission' service described in RFC9171 section 5.12.

3.5 BUNDLE NODE REGISTRATION CONSTRAINTS

3.5.1 All ipn scheme endpoints in which a node is registered shall be identified by EIDs whose node number is the node number common to all the source node IDs of non-anonymous bundles sourced by the node's BPA.

NOTE – This clause means that a node uses a single node number for all non-anonymous ipn scheme bundles that it sends but may use multiple service numbers. That node number is the same as is encoded in all the endpoints in which the node is registered.

3.5.2 No two BPAs shall register in endpoints whose EIDs have the same node number.

3.6 MINIMUM SUPPORTED BUNDLE SIZE

Conformant CCSDS implementations shall be able to forward and/or deliver bundles whose total size, including all extension blocks, is less than or equal to $10*2^{20}$ bytes (10 MB).

NOTE – Disposition of larger bundles is implementation-specific.

3.7 BUNDLE PROTOCOL SECURITY

Implementations of this specification are not required to implement BPSec (RFC9172).

4 SERVICE DESCRIPTION

4.1 SERVICES AT THE USER INTERFACE

4.1.1 The BP shall provide the following services to application(s):

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

b) terminate a registration;

c) switch a registration between Active and Passive states as discussed in RFC 9171;

d) transmit an ADU to an identified bundle endpoint;

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

f) receive an ADU contained in a delivered 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 data for one or more other transactions is still in progress. (But neither is support for this functional model precluded.)

4.1.3 Error indications at the service interface are implementation matters not covered by this specification.

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;

– Poll.request.

4.2.2 The BP service shall deliver the following indication primitives:

– BundleSendRequest.indication;

– BundleDelivery.indication.

4.3 SUMMARY OF PARAMETERS

4.3.1 DESTINATION ENDPOINT ID

The destination endpoint ID parameter shall identify the communication endpoint to which the bundle is to be delivered.

4.3.2 SOURCE NODE ID

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

NOTE – Source node IDs are singleton EIDs in which the node is registered as defined in RFC9171. In particular, when using the ipn URI scheme, the source node ID includes both a node number and a service number as described in 2.4.

4.3.3 REPORT-TO 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.

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

4.3.4 CREATION TIMESTAMP

The creation timestamp shall comprise the bundle creation time and the creation timestamp sequence number.

NOTE – Implementations may choose to manage a single, global timestamp sequence counter or manage individual timestamp sequence counters for disjunct sets of source node IDs . Sequence counters may be reset to zero when the current time advances by one millisecond. The combination of source node ID and bundle creation time stamp can serve as a unique ID for single bundle transmission request.

4.3.5 SEND REQUEST OPTIONS

4.3.5.1 The send request parameters shall indicate what optional procedures are additionally to be followed when transmitting the bundle and what optional services are requested.

4.3.5.2 The value of the send request parameters shall include the following:

a) ADU is an administrative record;

b) bundle must not be fragmented;

c) acknowledgement by application is requested;

NOTE – Information about requests for acknowledgement by applications is assumed to be passed to receiving applications when bundles are delivered. How applications respond to such requests is application-specific.

d) request reporting of bundle reception;

e) request reporting of bundle forwarding;

f) request reporting of bundle delivery;

g) request reporting of bundle deletion;

h) status time is requested in all status reports.

NOTE – Implementations may also allow inclusion of other information with the Send Request Parameters, such as metadata and material to be included, in particular, extension blocks.

4.3.6 BUNDLE DELIVERY INDICATION PARAMETERS

4.3.6.1 The delivery indication parameters shall be the ADU and the metadata from 4.3.6.2 below pertaining to the ADU.

4.3.6.2 The value of the delivery indications parameters shall include the following:

a) ADU is an administrative record;

b) acknowledgement by application is requested;

NOTE – Implementations may also include other information with the Bundle Delivery Indication Parameters such as the source EID, creation timestamp, and/or information from extension blocks.

c) acknowledgement by application requested flag;

4.3.7 LIFETIME PARAMETER

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

4.3.8 APPLICATION DATA UNIT PARAMETER

The ADU parameter shall indicate the ADU to be conveyed by the bundle.

4.3.9 BUNDLE ID

The Bundle ID parameter shall identify a particular bundle. The Bundle Send Request ID comprises the source node ID and creation timestamp.

4.3.10 DELIVERY FAILURE ACTION

4.3.10.1 The Delivery Failure Action parameter shall identify the response the node is required 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 9171 section 5.7 (Bundle Delivery) 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.' The delivery failure action associated with a registration 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 5.7 of RFC 9171.

4.3.12 BUNDLE DELIVERY METADATA

The Bundle Delivery Metadata parameter shall at minimum indicate the bundle's processing control flags, the destination endpoint ID, delivered bundle's remaining time to live, and the time the bundle was received.

4.4 BUNDLE PROTOCOL 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 (endpoint ID,

[default failure action])

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

Receipt of Register.request shall cause the BPA to declare the node's registration in the indicated endpoint.

NOTE – If the scheme of the indicated endpoint ID is ipn, then the node number of the indicated endpoint ID must be the node number of the BPA.

The registration shall initially be in Passive state.

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

Registration in particular endpoints (especially those associated with the node number of the node) may be implicit in the instantiation of the BPA or could require explicit registration requests from applications.

4.4.2 Deregister.request

4.4.2.1 Function

The Deregister.request primitive shall be used to notify the BPA 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 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

None.

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 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 BPA to deliver to the application all bundles destined for the indicated endpoint, for which delivery was deferred.

4.4.3.5 Discussion—Additional Comments

Changing the state of the registration to ‘active’ implicitly associates with that endpoint the application that issued the request. The expected effect of this association is that all bundles destined for this endpoint will be delivered to that application, but the details of this association are an implementation matter.

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 ADU 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 node ID,
                destination endpoint ID,
                report-to endpoint ID,
                send request 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 and shall cause a BundleRequestID.indication to be returned to the issuer of the send request.

4.4.4.5 Discussion—Additional Comments

None.

4.4.5 POLL.REQUEST

4.4.5.1 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.

4.4.5.2 Semantics

Poll.request shall provide parameters as follows:

Poll.request (destination communications endpoint ID)

4.4.5.3 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.

4.4.5.4 Effect on Receipt

Receipt of Poll.request shall cause the BPA to deliver to the BP application the least-recently received bundle destined for the destination communications EID, 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.

4.4.5.5 Discussion—Additional Comments

None.

4.4.6 BundleDelivery.indication

4.4.6.1 Function

The BundleDelivery.indication primitive shall be used to deliver the ADU and associated metadata to the service user.

4.4.6.2 Semantics

BundleDelivery.indication shall provide parameters as follows:

BundleDelivery.indication	(bundle ID, bundle delivery metadata, application data unit)

4.4.6.3 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.

4.4.6.4 Effect on Receipt

The effect on receipt is defined by the application.

4.4.6.5 Discussion—Additional Comments

None.

4.4.7 Send.indication

4.4.7.1 Function

The Send.indication primitive shall be used to provide information to a sending application about a bundle that the application caused to be created via a previous Send.request . Since the indication is a ‘bundle ID’, which contains the source EID and the bundle creation timestamp, it may not be generated immediately after the Send.request is received if the bundle implementation delays generating a bundle from the request. If the generation of the Send.indication is asynchronous with respect to the Send.request, some implementation-specific mechanism to associate the indication with the request that triggered it would be necessary. Such implementation-specific mechanisms are beyond the scope of this book.

4.4.7.2 Semantics

Send.indication shall provide parameters as follows:

Send.indication (bundle ID)

4.4.7.3 When Generated

Send.indication shall be generated by a BPA upon creation of a bundle in response to a Send.request primitive by the application.

4.4.7.4 Effect on Receipt

The effect on receipt is defined by the application.

4.4.7.5 Discussion—Additional Comments

None.

5 BUNDLE PROTOCOL NODE REQUIREMENTS

5.1 DISCUSSION

BP implements the mechanisms needed to create, forward, and receive bundles. To do so, it relies on the existence of services from some external source (e.g., the spacecraft on which the bundle node resides). This section lists the services that BP needs from some external source in order to function. It is broken into operational requirements (basic services such as storage and a source of time) and underlying communication service requirements (external services that effect transmission and reception).

5.2 OPERATIONAL REQUIREMENTS

5.2.1 BP nodes shall have access to a 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.

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

a) forward advancing time that can be represented as ‘DTN time’ as defined by RFC 9171 (reference [1]);

b) a counter conforming to the requirements of section 4.2.7 in RFC 9171 to provide sequence numbers for the creation timestamp fields of bundles.

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

5.3 UNDERLYING COMMUNICATION SERVICE REQUIREMENTS

5.3.1 Each convergence layer protocol adapter shall provide the following services to the BPA:

a) accepting a bundle from a bundle node that is reachable via the convergence layer protocol;

b) notifying the BPA of the disposition of its data sending procedures with regard to a bundle, upon concluding those procedures;

c) rendering to the BPA a bundle that was sent by a bundle node via the convergence layer protocol.

NOTES

1 The convergence layer service interface specified here is neither exhaustive nor exclusive. That is, supplementary DTN protocol specifications (including, but not restricted to, the BPSec as specified in RFC 9172) may expect CLAs 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.

2 Additionally, BP relies on the capabilities of protocols at the convergence layer to minimize congestion. The potentially long round-trip times characterizing delay-tolerant networks are incompatible with end-to-end reactive congestion control mechanisms, so convergence-layer protocols are expected to provide rate limiting or congestion control.

5.3.2 The service provided by the protocols beneath BP (not necessarily by the convergence layer protocol itself) shall deliver only complete bundles to the receiving BP node.

5.3.3 Render duplicate bundles to a BPA by the underlying layer shall be acceptable.

ANNEX A

PROTOCOL IMPLEMENTATION CONFORMANCE STATEMENT PROFORMA

(NORMATIVE)

A1 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.

The PICS states which capabilities and options of the protocol have been implemented. The following can use the PICS:

a) the protocol implementer, as a checklist to reduce the risk of failure to conform to the standard through oversight;

b) 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;

c) 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);

d) a protocol tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

A2 INSTRUCTIONS FOR COMPLETING THE REQUIREMENTS LIST

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:

a) CCSDS BP (this document);

b) RFC 9171 (reference [1]).

A4.2 In the tables below, the notation in the Reference column combines one of the short-form document identifiers above (e.g., RFC 9171) with applicable subsection numbers in the referenced document. RFC numbers are used to facilitate reference to subsections within the Internet specifications.

A5 GENERAL INFORMATION

A5.1 IDENTIFICATION OF PICS

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

A5.2 IDENTIFICATION OF IMPLEMENTATION UNDER TEST

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	

A5.3 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)	

A5.4 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.	a) Yes b) No
5	Date of statement (DD/MM/YYYY)	

A5.5 BASIC REQUIREMENTS

Item Number	Item	Protocol Feature	Reference	Status	Support
1	BP Formatting	Formats bundles as BPv7 per RFC 9171	This document: 3.1; RFC 9171 Section 4 except section 4.2.5.1 and section 4.4	M	
2	Previous Node Receive	Recognizes, parses, and acts on the previous node extension block	RFC 9171 section 4.4.1	M	
3	Previous Node Produce	Create previous node extension block	RFC 9171 section 4.4.1	O	
4	Bundle Age Receive	Recognizes, parses, and acts on the bundle age extension block	RFC 9171 section 4.4.2	M	
5	Bundle Age Produce	Create bundle age extension block	RFC 9171 section 4.4.2 Conditions: If bundle creation time = 0 If bundle creation time != 0	C M O	
6	Hop Count Receive	Recognizes, parses, and acts on the hop count extension block	RFC 9171 section 4.4.3	M	

Item Number	Item	Protocol Feature	Reference	Status	Support
7	Hop Count Produce	Create hop count extension block	RFC 9171 section 4.4.3	O	
8	BPv7	Identifies bundles as version 7 in the primary block	RFC 9171 section 9.2	M	
9	IPN_naming	Support for the ipn URI scheme	This document: 3.2.1; RFC 9171 section 4.2.5.1.2	M	
10	Null endpoint	Support for the null endpoint	This document: 3.2.2; RFC 9171 section 4.2.5.1.1	O	
11	IPN Node No	Use ipn node numbers assigned by SANA	This document: 3.2.3	M	
12	IPN Service No	Use ipn service numbers assigned by IANA/SANA	This document: 3.2.4	M	
13	Bundle Creation Metadata	Bundle creation timestamp and timestamp sequence number assigned when ADU is accepted for transmission	This document: 3.3.1	M	
14	Bundle Send Request	The combination of source node ID and creation timestamp shall be returned to the sending application	This document: 3.3.2	M	
15	Source Node ID	The source node IDs for all non-anonymous bundles' sources shall have the same node number	This document: 3.3.3	M	
16	Registration Constraints	All endpoints in which a node is registered shall have the same node number	This document: 3.4	M	
17	BPA Node Numbers	The node number is the same as is encoded in all the endpoints in which the node is registered	This document: 3.5.1	M	
18	BPA Endpoint Registration	No two BPAs shall register in endpoints whose EIDs have the same node number	This document: 3.5.2	M	

Item Number	Item	Protocol Feature	Reference	Status	Support
19	Minimum Bundle Size	Supports processing of bundles whose total size no less than $10*2^{20}$ bytes (10 MB)	This document: 3.6	M	
20	BPSec	BPSec is not required for implementations of BPv7.	BPSec, RFC 9172; RFC 9171 section 8; This document 3.7	O	
21	Service Interface	Supports the service interface in section 4	This document: section 4	M	
22	BP Node	Services that BP needs from an external source	This document: section 5	M	
23	TCP CLA	Implements bundle encapsulation in TCP segments	This document: B2.1.2	O.1	
24	LTP CLA	Implements bundle encapsulation in LTP blocks	This document: B2.1.4	O.1	
25	UDP CLA	Implements bundle encapsulation in UDP datagrams	This document: B2.1.3	O.1	
26	Space Packets CLA	Implements encapsulation of bundles in Space Packets	This document B2.1.5	O.1	
27	EPP CLA	Implements encapsulation of bundles in encapsulation packets	This document B2.1.6	O.1	
28	BP Managed Information	Implements the BP managed information described in annex C	This document, annex C	M	
29	BP Data Structures	Follows RFC 9171 rules for data structures	RFC 9171 Section 4.2	M	
30	Block Structures	Follows RFC 9171 rules for details in blocks	RFC 9171 Section 4.3	M	
31	Extension Blocks	Follows RFC 9171 rules for details in extension blocks	RFC 9171 Section 4.4	M	
32	Generation of Administrative Records	Follows RFC 9171 rules for generation of administrative records	RFC 9171 Section 5.1 (only requirement is “off by default”)	M	
33	Bundle Transmission	Follows RFC 9171 procedures for bundle transmission	RFC 9171 Section 5.2	M	

Item Number	Item	Protocol Feature	Reference	Status	Support
34	Forwarding Contraindicated	Follows RFC 9171 procedures when forwarding is contraindicated	RFC 9171 Section 5.3	M	
35	Forwarding Failed	Follows RFC 9171 procedures when forwarding a bundle fails	RFC 9171 Section 5.4	M	
36	Forwarding Failed – return to previous node	Follows RFC 9171 procedures when forwarding fails to forward bundle to previous node	RFC 9171 Section 5.4.2 Step 1	O	
37	Bundle Expiration	Follows RFC 9171 procedures when a bundle expires	RFC 9171 Section 5.5	M	
38	Bundle Reception	Follows RFC 9171 procedures when receiving a bundle	RFC 9171 Section 5.6	M	
39	Local Bundle Delivery	Follows RFC 9171 procedures when delivering a bundle to the AA	RFC 9171 Section 5.7	M	
40	Bundle Fragmentation Supported	Implementation supports fragmentation of bundles per RFC 9171	RFC 9171 Section 5.8	O	
41	Bundle Fragmentation Procedures	Follows RFC 9171 procedures when fragmenting a bundle	RFC 9171 Section 5.8 Condition: Mandatory if Item 31 is true.	C	
42	ADU Reassembly	Follows RFC 9171 procedures when reassembling an ADU	RFC 9171 Section 5.9	M	
43	Bundle Deletion – generation of bundle deletion status report	Follows RFC 9171 procedures when deleting a bundle	RFC 9171 Section 5.10 Step 1	O	
44	Bundle Deletion – removal of retention constraints	Follows RFC 9171 procedures when deleting a bundle	RFC 9171 Section 5.10 Step 2	M	
45	Discarding a Bundle with no remaining retention constraints	Follows RFC 9171 procedures when discarding a bundle	RFC 9171 Section 5.11	O	

Item Number	Item	Protocol Feature	Reference	Status	Support
46	Canceling a Transmission	Follows RFC 9171 procedures when canceling an initial transmission.	RFC 9171 Section 5.12	O	
47	Administrative Records	Formats administrative records per RFC 9171	RFC 9171 section 6.1 Condition: Mandatory if item 49 is true.	C	
48	Bundle Status Reports	Formats status reports per RFC 9171	RFC 9171 section 6.1.1 Condition: Mandatory if item 49 is true.	C	
49	Generating Administrative Records	Follows RFC 9171 procedures when generating an administrative record	RFC 9171 Section 6.2	O	

ANNEX B

CONVERGENCE LAYER ADAPTERS

(NORMATIVE)

B1 OVERVIEW

This annex describes various 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 BP may interact. This annex is in no manner comprehensive or rigorous but contains CCSDS-supported CLAs that have been demonstrated under various environments, requested to be included at the time of this writing, and appear applicable to CCSDS users.

B2 CONVERGENCE LAYER ADAPTERS

B2.1 AVAILABLE Convergence Layer ADAPTERS

The currently available CCSDS-supported CLAs adapt the LTP, TCP, UDP, SPP, and EPP.

B2.1.1 General

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

B2.1.2 Transmission Control Protocol Convergence Layer Adapter

When sending/receiving bundles using TCP at the convergence layer, bundles shall be sent over TCP according to the Delay-Tolerant Networking TCP Convergence Layer Protocol (reference [4]).

NOTE – IANA has allocated TCP port 4556 for the TCP CLA.

B2.1.3 User Datagram Protocol Convergence Layer Adapter

B2.1.3.1 User Datagram Protocol Maximum Bundle Transmission Size

The maximum size of a bundle that can be encapsulated in the UDP (reference [8]) CLA is 65,507 bytes.

B2.1.3.2 Bundle Encapsulation in User Datagram Protocol

Each bundle shall be encapsulated into one UDP datagram with no additional bytes.

NOTES

1 It is desirable that BP agents endeavor to send bundles of such a size as not to require fragmentation by the IP layer. In practice, this generally means keeping the size of the IP datagram (including the IP and UDP headers, plus the bundle) to no more than 1500 bytes.

2 IANA has allocated UDP port 4556 for the UDP CLA.

B2.1.3.3 User Datagram Protocol Port Number

All implementations should use UDP port 4556/UDP.

B2.1.3.4 Network Interactions

All implementations should ensure that the traffic sent by the UDP CLA does not adversely affect other traffic on the network.

NOTES

1 Network characteristics can best be managed on a closed network or a network with reserved bandwidth. Alternatively, congestion control procedures can be adopted as described in RFC 8085 (reference [G2]).

2 UDP does not provide any congestion control; UDP CLAs that may be used over large, shared networks like the Internet should take measures to ensure that they do not adversely affect other traffic on the network. One such measure would be to control the rate at which UDP datagrams are emitted from the CLA; another would be to define a Datagram Congestion Control Protocol (DCCP)-based CLA. (See RFC 7122 for more information.)

B2.1.4 Reliable Licklider Transmission Protocol Convergence Layer Adapter

The LTP CLA encapsulates bundles in LTP blocks.

B2.1.4.1 Discussion

LTP (reference [9]) provides service primitives for reliable transmission of client service from one LTP engine to another with the following service primitives and parameters:

Transmission.request (destination client service ID, destination LTP engine ID, client service data to send, length of the red-part of the data)

RedPartReception.indication (session ID, red-part bytes, indication as to whether or not the last byte of the red-part is also the last byte of the block, source LTP engine ID)

Where:

– Destination client service ID identifies the layer-(N+1) service to which the segment is to be delivered by the receiving LTP engine that is providing the N-layer service; for aggregations of BPv7 bundles as defined below this will be set to ‘4.’

– Destination LTP engine ID is the LTP engine ID of the LTP engine that is to be the receiver of data blocks.

– Client Service Data to Send is the client data to be transmitted.

– Length of the red-part of the data indicates the size of the part of the data which is to be transmitted reliably; for the reliable LTP CLA this will be set to the total length of the data to be sent as there will be no data to be sent unreliably.

– Session ID uniquely identifies a transmission session.

– Red-part bytes is the part of the client service data which has been sent reliably; for the reliable LTP CLA this will be all the data to be sent.

– Indication as to whether or not the last byte of the red-part is also the last byte of the block; for the reliable LTP CLA this will always be true.

– Source LTP engine ID is the LTP engine ID of the LTP engine that has transmitted the client service data.

B2.1.4.2 Bundles Formatted as Concise Binary Object Representation Byte Strings

An LTP CLA aggregation of bundles shall be defined as a sequence of CBOR byte strings, each encapsulating one serialized bundle (reference [10]).

NOTE – This aggregation scheme will result in the following encoding:

Sequence	Item #1			…	Item #N		
Headers	Head		Content		Head		Content
Fields	Major Type	Argument	Byte String		Major Type	Argument	Byte String
Values	2	Length of CBOR serialized Bundle #1	CBOR serialized Bundle #1		2	Length of CBOR serialized Bundle #N	CBOR serialized Bundle #N

B2.1.4.3 Bundle Encapsulation in Licklider Transmission Protocol

The reliable LTP CLA shall invoke the services of LTP by using the transmission.request with the following parameters:

– LTP client service ID shall be set to 4.

– LTP engine ID for the specific link destination

– LTP client service data shall be an LTP CLA aggregation of bundles as defined in B2.1.4.1.1.

B2.1.4.4 Bundle Reception for Bundle Encapsulated in Licklider Transmission Protocol

B2.1.4.4.1 A RedPartReception.indication shall cause the Reliable LTP CL to extract the individual bundles from the CBOR byte strings and provide those to the BPA.

B2.1.4.5 Licklider Transmission Protocol Client Service Identifier

SANA shall be requested to reserve LTP client service identifier '4' to identify an LTP CLA aggregation of bundles as defined in B2.1.4.1.

B2.1.5 Space Packet Protocol Convergence Layer Adapter

B2.1.5.1 Discussion

The SPP (reference [5]) provides service primitives for a PACKET and for an OCTET_STRING service. For BPv7, the OCTET_STRING service providing the following service primitives and parameters is recommended:

OCTET_STRING.request	(Octet String, APID, Secondary Header Indicator, Packet Type, Packet Sequence Count/Packet Name)
OCTET_STRING.indication	(Octet String, APID, Secondary Header Indicator, Data Loss Indicator (optional))

Where:

– Octet string is the service data unit transferred by the SPP.

– APID uniquely identify the source, destination, or type of the Space Packet.

– Secondary Header Indicator indicates the presence or absence of a Packet Secondary Header.

– Packet type is used to distinguish Packets used for telemetry (or reporting) from Packets used for telecommand (or requesting).

– Packet Sequence Count provides the sequential binary count of each Space Packet generated by the user application identified by the APID.

– Packet Name is only allowed for telecommand packets and will not be used for BPv7.

– (Optional) Data Loss Indicator may be used to alert the user in a destination end system that one or more Octet Strings have been lost during transmission, as evidenced by a discontinuity in the Packet Sequence Count.

In principle, the PACKET service can be used for BPv7 if BPv7 provides space packets to that service which are confirming to the following specifications.

B2.1.5.2 Space Packet Protocol Maximum Bundle Transmission Size

The maximum size of a bundle that can be transferred using the SPP CLA shall be 65,536 bytes.

B2.1.5.3 Bundle Encapsulation in Space Packet Protocol

The SPP CLA shall invoke the services of the SPP by using the Octet_String.Request with the following parameters:

– Octet string shall be a single CBOR serialized bundle

– Managed information shall be used to determine the APID

– Packet Secondary Header Indicator shall be set to absent

– Packet Sequence Count shall always be used instead of a Packet Name.

– The optional Data Loss Indicator shall be ignored.

B2.1.5.4 Bundle Reception for Bundles Encapsulated in SPP

An Octet_String.indication shall cause the SPP CLA to provide the included octet string to the BPA.

B2.1.6 Encapsulation Packet Protocol Convergence Layer

B2.1.6.1 Discussion

The EPP (reference [6]) provides the following service primitives and parameters:

ENCAPSULATION.request (Data Unit, SDLP_Channel, EPI)

ENCAPSULATION.indication (Data Unit, SDLP_Channel, EPI)

Where:

– Data Unit is the service data unit transferred by the EPP.

– SDLP_Channel is part of the Service Access Point (SAP) address of the EPP. It uniquely identifies the channel of the underlying Space Data Link Protocol (SDLP) through which the protocol data unit is to be transferred. Reference [6] describes the SDLP_Channel semantics; the exact semantics depend on the underlying SDLP services.

– EPI is part of the SAP address of the Encapsulation Service; it identifies the external protocol data unit to be encapsulated by this protocol.

B2.1.6.2 Encapsulation Packet Protocol Maximum Bundle Transmission Size

The maximum size of a bundle that can be transferred using the EPP CLA shall be 4,294,967,287 bytes.

B2.1.6.3 Bundle Encapsulation in Encapsulation Packet Protocol

The EPP CLA shall invoke the services of the EPP Encapsulation.request with the following parameters:

– Data unit shall be a single CBOR serialized bundle;

– Managed information shall be used to determine the SDLP_Channel;

– EPI value shall be set to the CCSDS Encapsulation Protocol Identifier ‘4’ as registered for BP in SANA (reference [7]).

B2.1.6.4 Bundle Reception for Bundles Encapsulated in Encapsulation Packet Protocol

An Encapsulation.indication shall cause the EPP CLA to provide the included data unit to the BPA.

ANNEX C

BUNDLE PROTOCOL MANAGED INFORMATION

(INFORMATIVE)

C1 OVERVIEW

It is recommended that the language of a standard for BP network management, not yet defined, conform to the canonical nomenclature defined in this annex. Managed information as defined and described in this annex provides a data model for use when implementing a management architecture. Formal terms, logical data types, and encoding will be provided in the BP network management standard.

C2 BASIC REQUIREMENTS

C2.1 Upon request, each BP node provides a set of managed information that represents the state of the node at a particular time.

C2.2 The minimal set of such information includes those data items identified by RFC 9171 and collected in this annex.

NOTE – The manner in which the information is requested and provided/delivered is an implementation matter.

C2.3 BP nodes 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.

C2.4 In addition to required information, each BP node may choose to provide supplementary information. Each identified managed information item specifies whether its collection and accurate reporting is required or recommended.

NOTES

1 In the future, managed information may be queried and delivered via a network management protocol.

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.

C3 BUNDLE STATE INFORMATION

C3.1 OVERVIEW

Bundles do not have a natural end state within a node; they are forwarded and/or delivered and/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.

C3.2 SUPPORTED TYPES OF BUNDLE STATE INFORMATION

BP nodes support the bundle state information itemized in table C-1.

Table C-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 Reassembly	The number of bundles/bytes associated with the retention constraint ***reassembly pending*** at this node.	Cumulative Bytes	No
		Cumulative Bundles	Yes
Counters			
Bundles Sourced	The number of bundles/bytes generated by this node.	Cumulative Bytes	No
		Cumulative Bundles	Yes
Bulk Bundles Queued	The number of bundles/bytes 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 node.	Cumulative Bundles	Yes

C4 NODE ERROR AND REPORTING INFORMATION

C4.1 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.

C4.2 SUPPORTED TYPES OF ERROR AND REPORTING INFORMATION

BP nodes support the error and reporting information itemized in table C-2.

Table C-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
Hop Count Deletions	The number of bundles deleted with the ***Hop limit exceeded*** 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 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

C5 REGISTRATION INFORMATION

C5.1 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.

C5.2 SUPPORTED TYPES OF REGISTRATION INFORMATION

BP nodes support the registration information itemized in table C-3.

Table C-3: Registration Information

Managed Information Item	Description	Req?
Identity Information		
EID	The EID of this registered endpoint. Note – Nodes may register a very large set of endpoints (e.g., ipn:3.*), therefore, having single entries may not be possible.	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

C6 CONVERGENCE LAYER INFORMATION

C6.1 OVERVIEW

To exchange bundles between two communicating nodes each node must have a set of managed information to configure and operate the convergence layer protocols. A set of defined managed information must be exchanged so that the communicating nodes can configure the convergence layers prior to the time of connection so they can interoperate. For example, to invoke the EPP convergence layer Encapsulation.request, the SDLP_Channel must be agreed to and exchanged by the operators of the communicating nodes.

NOTE – The formal definition and exchange mechanism of this managed information is a subject of a future book.

C7 NODE STATE INFORMATION

C7.1 OVERVIEW

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

C7.2 SUPPORTED TYPES OF NODE STATE INFORMATION

BP nodes support the node state information itemized in table C-4.

Table C-4: Node State Information

Managed Information Item	Description	Req?
Node State Identity Information		
Node Administrative EID	The EID that uniquely and permanently identifies this node's administrative endpoint.	Yes
BP Version Numbers	The number(s) of the version(s) of the BP 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 D

SECURITY, SPACE ASSIGNED NUMBERS AUTHORITY, AND PATENT CONSIDERATIONS

(INFORMATIVE)

D1 SECURITY

D1.1 OVERVIEW

The BP 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 BPSec 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 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.

D1.2 SECURITY CONCERNS WITH RESPECT TO THE CCSDS DOCUMENT

The BPv7 specification (reference [1]) contains a security section (8), which addresses necessary measures to protect BP data and recommends the use of BPSec of RFC 9172. Two types of security blocks are defined in RFC 9172:

a) 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 Block Confidentiality Block (BCB). Because security policy at BPSec nodes may differ regarding integrity verification, BIBs do not guarantee hop-by-hop authentication, as discussed in RFC9172 section 1.1.

b) Block Confidentiality Block (BCB) – Indicates that the security target(s) have been encrypted at the BCB security source 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.

c) This specification does not require adoption of RFC9172. Implementers are encouraged to utilize RFC9172 and/or the forthcoming CCSDS profile of it if they need security services. Because RFC9171 requires implementing RFC9172, an IETF-

compliant implementation could send bundles that use security services to a CCSDS BPv7 implementation, which might be unable to decrypt parts of those bundles.

D1.3 AUDITING OF RESOURCE USAGE

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

D1.4 POTENTIAL THREATS AND ATTACK SCENARIOS

No potential threat or attack scenarios are discussed.

D1.5 CONSEQUENCES OF NOT APPLYING SECURITY TO THE TECHNOLOGY

By not applying the native security of BP and the extended security of BPSec allowed by BP, the system must rely on security measures provided at the CLA interfaces and below. For space applications, these may be nonexistent or limited in capability due to 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.

D2 SPACE ASSIGNED NUMBERS AUTHORITY CONSIDERATIONS

SANA provides a node number registry that uses a space delegated to it by IANA for the registration of node numbers. While this registry is sufficient to prevent the unintentional reuse of node numbers across missions, it does not provide any information about the capabilities (e.g., CLAs, supported extension blocks, scheduled routing schedules, supported services) of specific nodes, including information about how to connect to such nodes.

To provide a link between sites supporting BP nodes and points of contact that can provide the information needed to communicate with the nodes, it is proposed to leverage the Service Sites and Apertures (SS&A) registry of SANA. For sites supporting BP services, the existing fields in the SS&A registry will be used to identify the node and the point of contact.

To support the linkage between node numbers and points of contact who can provide information about how to connect to those nodes, it is requested that SANA add a field to the Site Services portion of the SS&A that contains a list of the node numbers of the BP nodes at the site. Users should also be able to query the SS&A registry for the sites providing BP services.

It should be noted that the union of all of the node numbers mentioned in the various SS&A registry entries constitutes the complete set of CCSDS bundle nodes necessary for a user to

participate in the network. More specifically, agencies are expected to register any terrestrial BP infrastructure that might be used in cross-support activities in the SS&A registry.

This document also requests that SANA add a point of contact column to the CBHE node numbers registry for each allocated CBHE node range.

D3 PATENT CONSIDERATIONS

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

ANNEX E

BUNDLE PROTOCOL ELEMENT NOMENCLATURE

(INFORMATIVE)

E1 BUNDLE PROTOCOL BLOCK TABLES

This annex specifies the canonical nomenclature for DTN BPv7 block field definitions. In the 'Term' column, the non-canonical terms are given. The full canonical name is formed by prepending 'BPv7.' and the table name transformed into camelCase followed by a dot. For example, the full canonical name of the 'isFragment' field in the primary block is:

BPv7.primaryBlock.controlFlags.isFragment

This annex does not imply anything about implementation, encoding of values, or range limitations set by the encoding or implementation. (For encoding and limits set by the encoding methods, see RFC 9171.)

Value limits imposed by implementations will be documented by forthcoming network management specifications.

NOTE – It is recommended that the language of a standard for BP network management, as yet undefined, will conform to the canonical nomenclature defined in this annex.

E2 PRIMARY BLOCK ELEMENTS

Table E-1: Primary Block

Term		Logical Data Type	Range
bundleVersion		unsigned integer	(0 ..)
bundleControlFlags	**isFragment**	Boolean	(0 .. 1)
	isAdmin	Boolean	(0 .. 1)
	doNotFragment	Boolean	(0 .. 1)
	E2EAckRequested	Boolean	(0 .. 1)
	statusReportTimeRequested	Boolean	(0 .. 1)
	receivedStatusRequested	Boolean	(0 .. 1)
	forwardedStatusRequested	Boolean	(0 .. 1)
	deliveredStatusRequested	Boolean	(0 .. 1)

Term		Logical Data Type	Range
	deletedStatusRequested	Boolean	(0 .. 1)
crcType		unsigned integer	(0 .. 2)
destinationEID		EID	(Dependent on addressing scheme)
sourceEID		EID	(Dependent on addressing scheme)
reportToEID		EID	(Dependent on addressing scheme)
creationTimestamp	**bundleCreationTime**	unsigned integer	(0 ..)
	sequenceNumber	unsigned integer	(0 ..)
bundleLifetime		unsigned integer	(0 ..)
fragmentOffset		unsigned integer	(0 ..)
totalADULength		unsigned integer	(1 ..)
crcValue		byte string	(0 ..)

NOTES

1 The value of the primaryBlock.BundleVersion field for the version of the BP specified in this document is 7.

2 The fragmentOffset and totalADULength fields are only present if the bundle is a fragment.

E3 BLOCK SHARED ELEMENTS

All blocks other than the primary block share a common structure that includes information about the block, cyclic redundancy check (CRC) information, and a block content field. Those shared elements are represented in the table E-2.

NOTE – At the time of this specification, the following block types are defined:

- Payload Block: blockType Range (1);
- Previous Node Block: blockType Range (6);
- Age Block: blockType Range (7);
- Hop Count Block: blockType Range (10).

Table E-2: Block Metadata

Term		Logical Data Type	Range
blockType		unsigned integer	(0 ..)
blockNum		unsigned integer	(1 ..)
processingControlFlags	**replicateInAllBlocks**	Boolean	(0 .. 1)
	reportStatusIfUnprocessed	Boolean	(0 .. 1)
	deleteIfUnprocessed	Boolean	(0 .. 1)
	removeIfUnprocessed	Boolean	(0 .. 1)
crcType		unsigned integer	(0 .. 2)
blockContent		blockContentType	(Dependent on value of blockType)
crcValue		byte string	(0 ..)

E4 PAYLOAD BLOCK

Table E-3: Payload Block

Term		Logical Data Type	Range
blockContentType	payload	byte string	NA

E5 PREVIOUS NODE BLOCK

Table E-4: Block Content for Previous Node Block

Term		Logical Data Type	Range
blockContentType	eidForwarded	EID	(Dependent on addressing scheme)

E6 BUNDLE AGE BLOCK

Table E-5: Block Content for Bundle Age Block

Term		Logical Data Type	Range
blockContentType	bundleAge	unsigned integer	(0..2^64-1)

E7 HOP COUNT BLOCK

Table E-6: Block Content for Hop Count Block

Term		Logical Data Type	Range
blockContentType	bundleHopLimit	unsigned integer	(1 .. 255)
	bundleHopCount	unsigned integer	(0 .. 255)

E8 ADMINISTRATIVE RECORD

Table E-7: Administrative Record

Term		Logical Data Type	Range
adminRecordStructure	recordType	unsigned integer	(0..2^64-1)
	recordContent	Variant type (see note 1)	(Dependent on recordTypeCode)

NOTE – At the time of this specification, the following record types are defined:

Bundle Status Report: RecordType Range[1]

[1] Variant type dependent on the value of recordTypeCode. RFC 9171 defines a recordContent for Bundle Status Record (BSR).

E9 BUNDLE STATUS REPORT ADMINISTRATIVE RECORD CONTENT

Table E-8: Record Content for Bundle Status Report

Term			Logical Data Type	Range
BSRRecordContentType	BSRStatus		BSRStatusType	(See below - BSRStatusType)
	BSRReasonCode		unsigned integer (see note 2)	(0..2^64-1)
	subjectSourceEID		EID	(Dependent on addressing scheme)
	subjectCreation Timestamp	bundleCreationTime	unsigned integer	(0..)
		sequenceNumber	unsigned integer	(0..)
	subjectFragmentOffset (see note 4)		unsigned integer	(0..2^64-1)
	subjectTotalADULength (see note 4)		unsigned integer	(0..2^64-1)
BSRStatusType	receivedEvent		eventDataPointType	(See below - eventDataPointType)
	forwardedEvent		eventDataPointType	(See below - eventDataPointType)
	deliveredEvent		eventDataPointType	(See below - eventDataPointType)
	deletedEvent		eventDataPointType	(See below - eventDataPointType)
eventDataPointType	eventAssertion		Boolean	(0 .. 1)
	eventTimestamp (see note 5)		unsigned integer (see note 3)	(0..2^64-1)

NOTES

1 Administrative records are carried as payloads of bundles and are signaled by the BPv7.primaryBlock.bundleControlFlags.isAdmin field.

2 Enumerated values form the set of Valid status report reason codes that are registered in the IANA ‘Bundle Status Report Reason Codes’ subregistry in the ‘Bundle Protocol’ registry.

3 Unsigned integer represents the DTN Time.

4 This is optional and is present if and only if the bundle whose status is being reported was a fragment.

5 This is optional and is present if the eventAssertion is 1 AND the ‘Report status time’ flag was set to 1 in the bundle processing control flags of the bundle whose status is being reported.

ANNEX F

INTERPLANETARY NETWORK UNIFORM RESOURCE IDENTIFIER SCHEME UPDATES

(INFORMATIVE)

This document references the ipn URI scheme per RFC9171, where endpoint identifiers are of the form <node number>.<service number>. The IETF DTN WG is preparing an update to the ipn URI scheme to include an optional naming authority so that fully qualified ipn EIDs could be of the form <authority>.<node_number>.<service number>. The existing format (<node_number>.<service_number>) and CBHE node range allocated to SANA are expected to remain valid.

ANNEX G

INFORMATIVE REFERENCES

(INFORMATIVE)

[G1] *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.

[G2] L. Eggert, G. Fairhurst, and G. Shepard. UDP Usage Guidelines. RFC 8085. Reston, Virginia: ISOC, March 2017.

[G3] E. Birrane and K. McKeever. *Bundle Protocol Security (BPSec)*. RFC 9172. Reston, Virginia: ISOC, January 2022.

[G4] E. Annis, E. Birrane, and S. Heiner. *Delay-Tolerant Networking Management Architecture.* RFC 9675. Reston, VA: ISOC, November 2024. https://datatracker.ietf.org/doc/rfc9675/

ANNEX H

IMPLEMENTATION AND TESTING

(INFORMATIVE)

European Space Agency Bundle Protocol

ESA developed and currently maintains a full operational implementation of the BPv7 for the ground segment. ESA BP is a Java stand-alone application developed for Ground Systems that fully supports all features in this specification. It provides convergence layers for many standard protocols, such as TCP, EPP, SPP, AOS, TC, TM, LTP, and Space Link Extension, and it is deployed in the ESA operational network. ESA BP has been used for on-orbit DTN demonstrations [1] and is available under ESA Community Licence in the space CODEV platform [2].

[1] C. Malnati and F. Flentge, "DTN Demonstrations with ESA Ground Segment," in IEEE Journal of Radio Frequency Identification, vol. 8, pp. 609-617, 2024, doi: 10.1109/JRFID.2024.3415746.

[2] Space CODEV Platform: https://www.space-codev.org/

Unibo-BP

Unibo-BP is a BP implementation designed and maintained by the University of Bologna. It is written in C++, research-driven, space-oriented, and fully compliant with RFC 9171. Unibo-BP is not a stand-alone application, but the core of a wide ecosystem that includes DTNsuite applications, LTP and TCPCLv3 convergence layers, and CGR/SABR routing. A comprehensive description of Unibo-BP can be found in [1]. Code, released as free software under the GPLv3 licence, can be downloaded from [2].

[1] C. Caini and L. Persampieri, "Design and Features of Unibo-BP, the Unibo Implementation of the DTN Bundle Protocol," in IEEE Journal of Radio Frequency Identification, vol. 8, pp. 458-467, 2024, doi: 10.1109/JRFID.2024.3358012.

[2] Unibo-BP code website: https://gitlab.com/unibo-dtn/unibo-bp

NASA DTN

NASA has several DTN implementations, which conform to the requirements in this Experimental Specification for BPv7.

NASA Implementation	**Repository**
BP cFS	https://github.com/nasa/bp
BPLib	https://github.com/nasa/bplib
DTNME	https://github.com/nasa/DTNME
HDTN	https://github.com/nasa/HDTN
ION	https://github.com/nasa-jpl/ION-DTN
ION-Core	https://github.com/nasa-jpl/ion-core
ION-Core/FPrime	https://github.com/fprime-community/fprime-dtn

ANNEX I

ABBREVIATIONS AND ACRONYMS

(INFORMATIVE)

Term	Meaning
AA	application agent
ADM	asynchronous data model
ADU	Application Data Unit
APID	application process identifier
AE	administrative element
DTNMA	DTN management architecture
AOS	Advanced Orbiting Systems
ASE	application-specific element
BCB	block confidentiality block
BIB	bundle integrity block
BP	Bundle Protocol
BPv7	Bundle Protocol Version 7
BPA	bundle protocol agent
BPSec	Bundle Protocol Security
BSR	Bundle Status Record
CBHE	Compressed Bundle Header Encoding
CBOR	Concise Binary Object Representation
CCSDS	Consultative Committee for Space Data Systems
CL	convergence layer
CLA	convergence layer adapter

CRC	cyclic redundancy check
DCCP	Datagram Congestion Control Protocol
DTKA	delay-tolerant key administration
DTN	delay/disruption tolerant networking
EID	endpoint identifier
EPI	EPP Protocol Identifiers
EPP	Encapsulation Packet Protocol
IANA	Internet Assigned Numbers Authority
IEC	International Electrotechnical Commission
IETF	Internet Engineering Task Force
ION	Interplanetary Overlay Network
IP	Internet Protocol
ipn	Interplanetary Internet
ISO	International Organization for Standardization
ISOC	Information Security Operations Center
LOS	loss of signal
LTP	Licklider Transmission Protocol
OSI	Open Systems Interconnection
PDU	protocol data unit
PICS	protocol implementation conformance statement
POC	point of contact
RFC	Request for Comments
RL	requirements list
SABR	Schedule Aware Bundle Routing

SANA	Space Assigned Numbers Authority
SAP	Service Access Point
SDLP	Space Data Link Protocol
SIS	Space Internetworking Services
SPP	Space Packet Protocol
SSP	scheme-specific part
SS&A	service site and apertures
TC	Telecommand
TCP	Transmission Control Protocol
TM	Telemetry
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
USLP	Unified Space Link Protocol
WG	working group