CCSDS
The Consultative Committee for Space Data Systems

Report Concerning Space Data System Standards

# RATIONALE, SCENARIOS, AND REQUIREMENTS FOR DTN IN SPACE

INFORMATIONAL REPORT

CCSDS 734.0-G-1

GREEN BOOK
August 2010

The Consultative Committee for Space Data Systems

Report Concerning Space Data System Standards

# RATIONALE, SCENARIOS, AND REQUIREMENTS FOR DTN IN SPACE

INFORMATIONAL REPORT

CCSDS 734.0-G-1

GREEN BOOK

August 2010

# AUTHORITY

| | |
|---|---|
| Issue: | Informational Report, Issue 1 |
| Date: | August 2010 |
| Location: | Washington, DC, USA |

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and reflects the consensus of technical panel experts from CCSDS Member Agencies. The procedure for review and authorization of CCSDS Reports is detailed in the *Procedures Manual for the Consultative Committee for Space Data Systems*.

This document is published and maintained by:

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

# FOREWORD

This document is a CCSDS Informational Report containing background and explanatory material to support forthcoming CCSDS Recommendations for a Delay-Tolerant Networking protocol suite.

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Report is therefore subject to CCSDS document management and change control procedures, which are defined in the *Procedures Manual for the Consultative Committee for Space Data Systems*. 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'Etudes Spatiales (CNES)/France.
- China National Space Administration (CNSA)/People's Republic of China.
- Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR)/Germany.
- European Space Agency (ESA)/Europe.
- Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
- Japan Aerospace Exploration Agency (JAXA)/Japan.
- National Aeronautics and Space Administration (NASA)/USA.
- Russian Federal Space Agency (RFSA)/Russian Federation.
- UK Space Agency/United Kingdom.

Observer Agencies

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

# DOCUMENT CONTROL

| Document | Title | Date | Status |
|---|---|---|---|
| CCSDS 734.0-G-1 | Rationale, Scenarios, and Requirements for DTN in Space, Informational Report, Issue 1 | August 2010 | Original issue |

# CONTENTS

Section Page

## CONTENTS (continued)

Section Page

Figure

## CONTENTS (continued)

Figure Page

Table

# 1 INTRODUCTION

## 1.1 PURPOSE

This document has been developed to present the concept and rationale for a CCSDS Recommended Standard for the Delay/Disruption Tolerant Networking (DTN) service. Specifically it describes the rationale, scenarios/use cases, and requirements for a proposed DTN service targeted at the space internetworking environment. While this document briefly reviews the benefits of networked communication, it takes as a starting point the desire for such a service. This document does not attempt to describe all the details of how a DTN service could be implemented and/or used; however, this Report will assist decision-makers and implementers with evaluating the applicability of such a service to mission needs.

## 1.2 SCOPE

This Report provides supporting descriptive and tutorial material for a DTN service. This document is not part of the Recommended Standard specifying such a service. In the event of conflicts between this Report and the Recommended Standard, the Recommended Standard shall prevail.

## 1.3 APPLICABILITY

This document can serve as a reference for mission designers considering the needs of their missions for DTN services, mission operations personnel determining what capabilities of DTN to invoke, protocol designers implementing DTN protocols, and other agency personnel structuring cross-support agreements.

## 1.4 RATIONALE

A set of CCSDS Recommended Standards will specify the DTN services and protocol mechanisms to implement those services. This document is intended to provide background and context for the Recommended Standards.

## 1.5 DOCUMENT STRUCTURE

Section 2 presents the overview and rationale for a DTN-based space internetworking service.

Section 3 describes a number of scenarios to motivate particular capabilities that the space internetworking protocol should support.

Section 4 draws from section 3 to derive a set of the required characteristics and capabilities of a space internetworking service.

Section 5 describes candidate technologies for deployment as a space internetworking service, including custom forwarding, the use of CCSDS Space Packets as a Network layer, the use of CCSDS File Delivery Protocol (reference [7]) store-and-forward overlay, and the Bundle Protocol (references [14] and [15]) as specified by the Delay Tolerant Networking Research Group (DTNRG) within the Internet Research Task Force (IRTF).

Section 6 concludes the document.

A number of annexes follow that address issues related to the deployment of and transition to an internetworked architecture as recommended by this document. Issues addressed include the coexistence of network PDUs with current packet mechanisms on space links, examples of how higher-layer services could be implemented over the proposed internetwork layer, and how the internetworking service can support remote, in-situ networks that use alternate protocols internally.

## 1.6 CONVENTIONS AND DEFINITIONS

### 1.6.1 GENERAL

Internetworking—constructing a more far-reaching network by defining a protocol layer that supports end-to-end delivery of data across multiple, possibly heterogeneous Data Link layer technologies.

### 1.6.2 DEFINITIONS FROM OSI BASIC REFERENCE MODEL

This Report makes use of a number of terms defined in reference [1]. The uses of those terms in this Report are to be understood in a generic sense, i.e., in the sense that those terms are generally applicable to any of a variety of technologies that provide for the exchange of information between real systems. Those terms are:

– entity;

– Protocol Data Unit (PDU);

– service;

– Service Access Point (SAP);

– Service Data Unit (SDU).

### 1.6.3 TERMS DEFINED IN THIS REPORT

**1.6.3.1** Internet Protocols—The protocols and services that are commonly used to implement and support internetworking via the Internet Protocol (IP) version 4 (reference [12]) or version 6 (reference [13]). Examples of supporting protocols include

Ethernet, IP Security (IPSec), the Domain Name Service (DNS), Simple Network Management Protocol (SNMP), and various routing protocols.

**1.6.3.2** DTN Protocols—The protocols and services used to implement and support internetworking via the Bundle Protocol (BP). Examples of supporting protocols and services include the Licklider Transport Protocol (LTP), the Bundle Security Protocol, and Contact Graph Routing (CGR).

**1.6.3.3** DTN Node (DTN Router)—A protocol entity that understands and processes the PDUs of the Bundle Protocol.

**1.6.3.4** Bundle—The PDU of the Bundle Protocol.

**1.6.3.5** Store-and-forward—The ability of the communications protocol to store data before it is forwarded. Unlike the Internet model of store-and-forward, where data is stored long enough to look up a next hop and queue the data on an outbound link, the consideration here is the possibility of storing PDUs for arbitrary lengths of time, even if no outbound path is currently available.

**1.6.3.6** Custody transfer—A network service that provides reliability on a hop-by-hop basis rather than an end-to-end basis. With custody transfer, the DTN node responsible for retransmitting data if that data is lost progresses towards the data destination.

**1.6.3.7** Custody transfer acknowledgement—The Bundle Protocol signaling that advances the point of retransmission for Bundle PDUs.

## 1.7 REFERENCES

The following documents are referenced in this Report. At the time of publication, the editions indicated were valid. All documents are subject to revision, and users of this Report 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 documents.

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

[2] *AOS Space Data Link Protocol*. Recommendation for Space Data System Standards, CCSDS 732.0-B-2. Blue Book. Issue 2. Washington, D.C.: CCSDS, July 2006.

[3] *TC Space Data Link Protocol*. Recommendation for Space Data System Standards, CCSDS 232.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2003.

[4] *TM Space Data Link Protocol*. Recommendation for Space Data System Standards, CCSDS 132.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2003.

[5] *Proximity-1 Space Link Protocol—Data Link Layer*. Recommendation for Space Data System Standards, CCSDS 211.0-B-4. Blue Book. Issue 4. Washington, D.C.: CCSDS, July 2006.

[6] *Mission Operations Message Abstraction Layer*. Recommendation for Space Data System Standards, CCSDS 521.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, July 2010.

[7] *CCSDS File Delivery Protocol (CFDP)*. Recommendation for Space Data System Standards, CCSDS 727.0-B-4. Blue Book. Issue 4. Washington, D.C.: CCSDS, January 2007.

[8] *CCSDS File Delivery Protocol (CFDP)—Part 1: Introduction and Overview*. Report Concerning Space Data System Standards, CCSDS 720.1-G-3. Green Book. Issue 3. Washington, D.C.: CCSDS, April 2007.

[9] *Space Packet Protocol*. Recommendation for Space Data System Standards, CCSDS 133.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2003.

[10] S. Burleigh. “Dynamic Routing for Delay-Tolerant Networking in Space Flight Operations.” In *Proceedings of the Eleventh International Conference on Space Operations (SpaceOps 2008)* (Heidelberg, Germany, May 12-16, 2008 ). AIAA-2008-3406. Reston, Virginia: AIAA, 2008.

[11] *Recommendations on a Strategy for Space Internetworking*. Report of the Interagency Operations Advisory Group Space Internetworking Strategy Group. N.p.: SISG, November 15, 2008.

[12] J. Postel. *Internet Protocol*. STD 5. Reston, Virginia: ISOC, September 1981.

[13] S. Deering and R. Hinden. *Internet Protocol*, *Version 6 (IPv6) Specification*. RFC 2460. Reston, Virginia: ISOC, December 1998.

[14] K. Scott and S. Burleigh. *Bundle Protocol Specification*. RFC 5050. Reston, Virginia: ISOC, November 2007.

[15] V. Cerf, et al. *Delay-Tolerant Networking Architecture*. RFC 4838. Reston, Virginia: ISOC, April 2007.

[16] S. Burleigh. *Compressed Bundle Header Encoding (CBHE)*. Internet-Draft. Reston, Virginia: ISOC, April 9, 2009.

[17] *Cross Support Reference Model—Part 1: Space Link Extension Services*. Recommendation for Space Data System Standards, CCSDS 910.4-B-2. Blue Book. Issue 2. Washington, D.C.: CCSDS, October 2005.

[18] *Space Link Extension—Return All Frames Service Specification*. Recommendation for Space Data System Standards, CCSDS 911.1-B-2. Blue Book. Issue 2. Washington, D.C.: CCSDS, December 2004.

[19] *Space Link Extension—Return Channel Frames Service Specification*. Recommendation for Space Data System Standards, CCSDS 911.2-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, December 2004.

[20] R. Gladden, et al. “Mars Relay Coordination Lessons Learned.” In *Proceedings of the IEEE Aerospace Conference, 2005* (Big Sky, MT, 5-12 March 2005), 177-190. New York: IEEE, 2005.

[21] *Spacecraft Onboard Interface Services—File and Packet Store Services*. Draft Recommendation for Space Data System Standards, CCSDS 873.0-R-2. Red Book. Issue 2. Washington, D.C.: CCSDS, July 2010.

[22] T. Berners-Lee, R. Fielding, and R. Fielding. *Uniform Resource Identifier (URI): Generic Syntax*. STD 66. Reston, Virginia: ISOC, January 2005.

# 2 OVERVIEW

## 2.1 OVERVIEW

The primary goal of CCSDS is to increase the level of interoperability between space organizations. Today, mission communication architectures are essentially point-to-point between the mission control center and the spacecraft. Standardization of a suite of cross-support services on the ground has extended and is continuing to extend this model so that agencies can share resources such as ground stations for cross support. This sharing is implemented by providing a standardized space link service interface (references [18] and [19]) at the ground station that accepts frames (and in the future, packets) for uplink and demultiplexes downlinked frames and delivers them to control centers using IP-based protocols.

This communication model has worked fine for a long period of time; however, as the number of space assets grows, and missions become more demanding, the communications architecture will become even more complex. In some instances it will be desirable to provide extra network ‘hops’ both in space and on the ground instead of using only a single data link between the mission control center and the spacecraft. Relays, whether they are spacecraft or ground stations, need to buffer data that cannot be transferred end-to-end because of visibility constraints, provide points for signal regeneration, switch Data Link layers to match the environment, and serve as decision points for data forwarding (routing). Today’s communications architecture will be hard-pressed to support these needs. It would become labor intensive, driving up the cost of operations. It imposes the risk of human error, which requires mitigation strategies that add cost. It is program limiting since cost and risk grow as the number of links and cross-links increase.

Take for example, the Mars Relay Operations scenario, shown in figure 2-1. This illustrates how space relays were used to communicate between the rovers on Mars and ground stations on Earth. Management of each of the links was complex and time-consuming. The different spacecraft were commanded with different timescales for planning and different planning horizons. Landers were typically operated daily, while orbiters were scheduled for weeks or months at a time. Iterated long- and short-term planning processes were needed to coordinate communications among the various landers and orbiters. These involved both monthly and weekly meetings, and automated conflict resolution software was extremely useful (reference [20]).

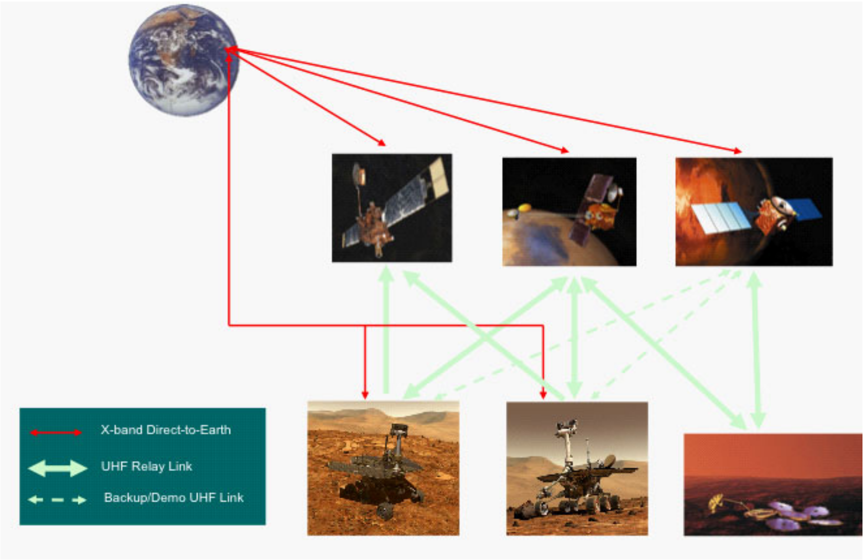


**Figure 2-1: Mars Relay Operations**

To address this concern, CCSDS proposes a space internetworking architecture that will allow different agencies to share extra-terrestrial (in space and at other planets) resources and to provide cross support to one another, even if the end systems are not directly accessible from the Earth. A common space internetwork design will support interoperability and lower design cost which, in turn, will allow resource sharing and the opportunity for greater science return and reduced mission risk.

The internetworking capabilities should support fully networked interoperability as shown in the following figure.

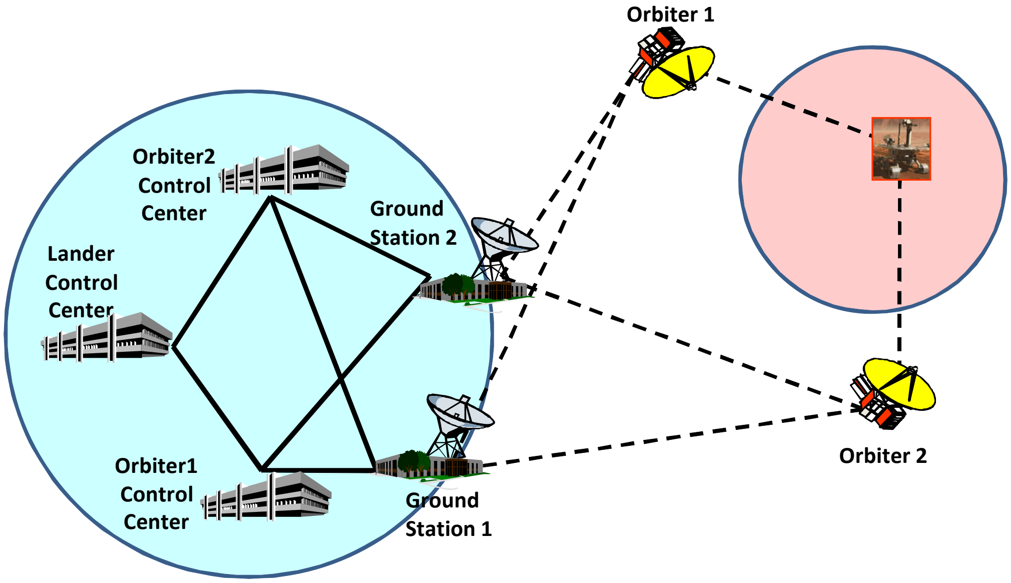


**Figure 2-2: Possible Data Paths in a Cross-Supported, Networked Architecture**

Typically a Network layer end-to-end data structure is used when data needs to be transferred across possibly heterogeneous links. This end-to-end data structure allows for logical addressing of the endpoints independent of the Data Link layer addresses and has some multiplexing mechanism to support higher-layer protocols. CCSDS currently recommends three data structures that could serve as end-to-end Network layer protocols: the CCSDS Space Packet Protocol, the Space Communications Protocol Specification—Network Protocol (SCPS-NP), and the Internet Protocols (IPv4/IPv6). Annex A discusses the following candidate networking technologies in more detail:

- custom forwarding (Application layer forwarding);
- CCSDS Space Packets as an internetworking packet format;
- CCSDS File Delivery Protocol (CFDP);
- Internet Protocol (IP);
- the Bundle Protocol from DTN.

Annex A describes each of these candidate technologies in detail. However, of the protocols mentioned, the Bundle Protocol associated with DTN seems best suited for the widest range of space internetworking environments. Like IP, the Bundle Protocol provides an internetwork-layer data unit with end-to-end addressing capabilities. Unlike IP, however, the Bundle Protocol does not assume continuous connectivity and specifically allows for in-network data storage such as might take place when Earth can transmit to an orbiter which then has to hold on to the data until the orbiter can relay the data to a landed asset. It is recognized that different mission requirements will probably drive development of parallel architectures, at least in parts of the design space, where some subset of the above mechanisms may all coexist.

## 2.2 CURRENT MISSION ARCHITECTURE

In the twentieth century, science and exploration spacecraft were built to communicate primarily with ground stations, with commands flowing from ground control center to spacecraft, and telemetry and data flowing from spacecraft to ground. There were few cases where a science spacecraft would communicate directly with another spacecraft or with multiple control centers on the ground. This situation is illustrated in figure 2-3.

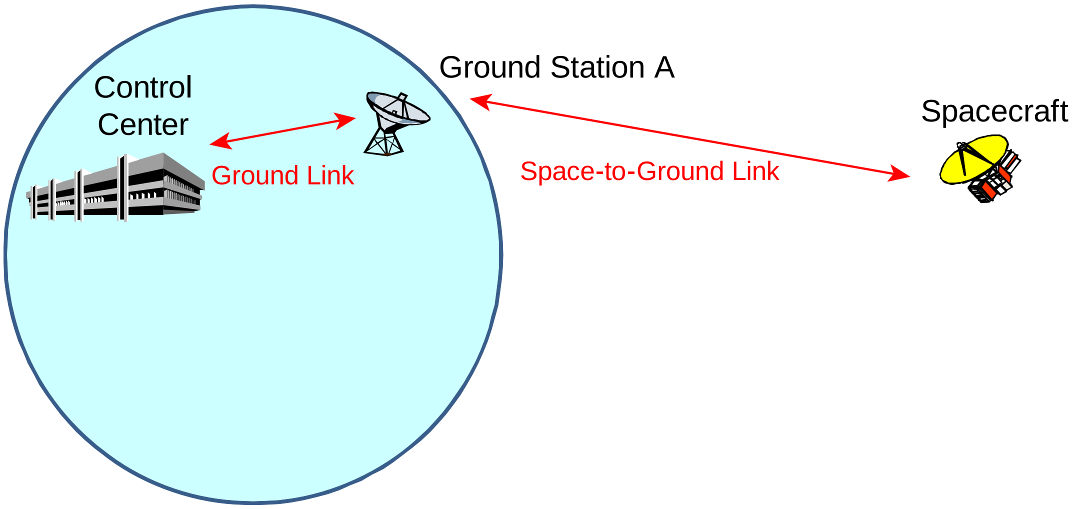


**Figure 2-3: Traditional Point-to-Point Space Mission Architecture**

This approach was successful and has supported many missions, but the data architecture that has evolved provides limited support for multi-hop networking because

a) some CCSDS Recommended Standards allow for optional capabilities that can result in incompatible implementations;

b) data structures have been optimized for managed, point-to-point communications.

The first problem can be addressed by modifying existing Recommended Standards and/or by constructing 'profiles' that restrict the protocol options in particular mission groups such as Mars relay operations. The second problem requires development of a new protocol or suite of protocols that better supports automated multi-hop forwarding of data.

## 2.3 RECENT ADVANCES

Experience with data relaying at Mars has demonstrated a number of advantages over traditional direct-to-Earth communications. These include:

– Increased science data return—The Mars Exploration Rovers (MERs) have used data relaying to increase data return substantially, from ~30Mb/sol achievable with the Direct-To-Earth (DTE) link to ~100-200Mb/sol via the Mars Odyssey orbiter.

– Lower power required—The MER DTE links require roughly 5 Watt hours per Mb of data return, while the relay uses around 0.1 Watt hour per Mb. This enables small scout-class mission concepts and increases the amount of energy available for science.

– Lower mass required—Relay operations require lower mass on landed elements, which are typically more mass-constrained than orbiters.

- More communications opportunities—Relaying typically supports more communications opportunities than DTE links. This in turn supports complex in-situ missions that might want to execute multiple command/telemetry cycles per sol.

These advantages are direct results of using in-space relaying instead of DTE data transfers from the rovers. Because the orbiting relays use different Data Link layers for Mars surface-to-orbit communications than for the Mars orbit-to-Earth links, they provide different data link services that are better suited to the local environments. For orbiter-to-Mars-surface communications, Proximity-1 (reference [5]) can be used in its reliable mode since round trip times are small and Automatic Repeat Request (ARQ) is both feasible and efficient. Reliability ensures that important data is successfully transferred before moving on to less important data. This reliability cannot be performed between the Mars surface and Earth because of the much longer round trip times. For the long-haul Mars-orbiter-to-Earth link, traditional telecommand and telemetry, including more powerful coding, are used.

As described above, current relay operations at Mars implement multi-hop relaying without true internetworking. There is no network-wide addressing scheme, no provision for different classes of data, and no true end-to-end Network layer data unit. These deficiencies will inhibit operations as more elaborate missions involving orders-of-magnitude-more systems and communication links, as well as human crews, are developed.

## 2.4 BENEFITS OF NETWORKED COMMUNICATIONS

The data relay benefits described above in the context of the MER missions are an example of benefits achievable within a single agency. Standardizing the relay (Network layer) protocols will enable the same types of cross support in space that are currently possible on the ground, with the additional benefits of signal regeneration at the relays, switching Data Link layers to suit the local environment, and the ability to make routing decisions both in space and on the ground.

The Space Internetworking Strategy Group (see reference [11]) chartered by the Inter-Agency Operations Advisory Group (IOAG) has come to similar conclusions when examining the current and future states of space internetworking. In particular, the Space Internetworking Strategy Working Group (SISG) report (reference [11]) makes the following recommendations:

- Recommendation R-1: There should be international agreement on how to do space-to-space interoperability and space-based infrastructure that supports space-to-space interoperability in a standard way.
- Recommendation R-2: In-space internetworking should be as fully verified as feasible in long delay mission environments.
- Recommendation R-3: There should be international agreement on how to manage space-to-space or end-to-end interoperability.

– Recommendation R-4: There should be interoperable services for timing, positioning, management, etc., in addition to services for relaying data.

## 2.5 FUTURE MISSIONS

Planning is also underway for missions that envision multiple nodes that communicate not only between space and ground but also among systems in space. Managing the connectivity and data transfers among this increasing number of systems will become more and more difficult. The situation is reminiscent of the early days of telephones and switchboards. When the number of systems was sufficiently small, human circuit switching with operators in the loop was possible. As the number of users grew, the phone system had to evolve to automated switching systems that were fully computer controlled and software upgradeable. The future space communication architecture requires a similar shift from traditional circuit-switched space communication toward a more flexible network architecture for space communication.

## 2.6 NEXT STEPS IN NETWORKING

By standardizing a multi-hop relay mechanism, CCSDS member agencies will lay one part of the technical foundation for interoperability and cross support in space.

By developing a set of Delay and Disruption services to be provided in subsequent documents, common data handling functions can be implemented in standard and hopefully reusable software/hardware. Moving these capabilities into the infrastructure allows mission software to focus on mission-specific functions instead of ‘re-inventing the wheel’ with each mission when it comes to communications. Finally, a common set of protocols for space-based internetworking will enable inter-agency cross support, which should increase science data return and decrease mission risk.

Relay operations will depend on interoperability at the lower layers of the communications stack such as the Physical and Data Link layers for compatible frequencies, modulation schemes, coding, etc. Thus one recommendation of this document is to further specify the protocols and protocol options needed for interoperability of space data links.

Given the above motivation, a number of protocols could be used to support multi-hop internetworking, including the Internet Protocol (IP) suite, CFDP, CCSDS Space Packets, and DTN.

## 2.7 SIS-DTN DOCUMENT MAP

Figure 2-4 illustrates the relationship among the documents this working group believes should be developed as part of an overall system to implement space internetworking and related documents produced by the IETF, including the IRTF's Delay Tolerant Networking Research Group. As of this time the group has only been chartered to produce those documents in the circle on the left.

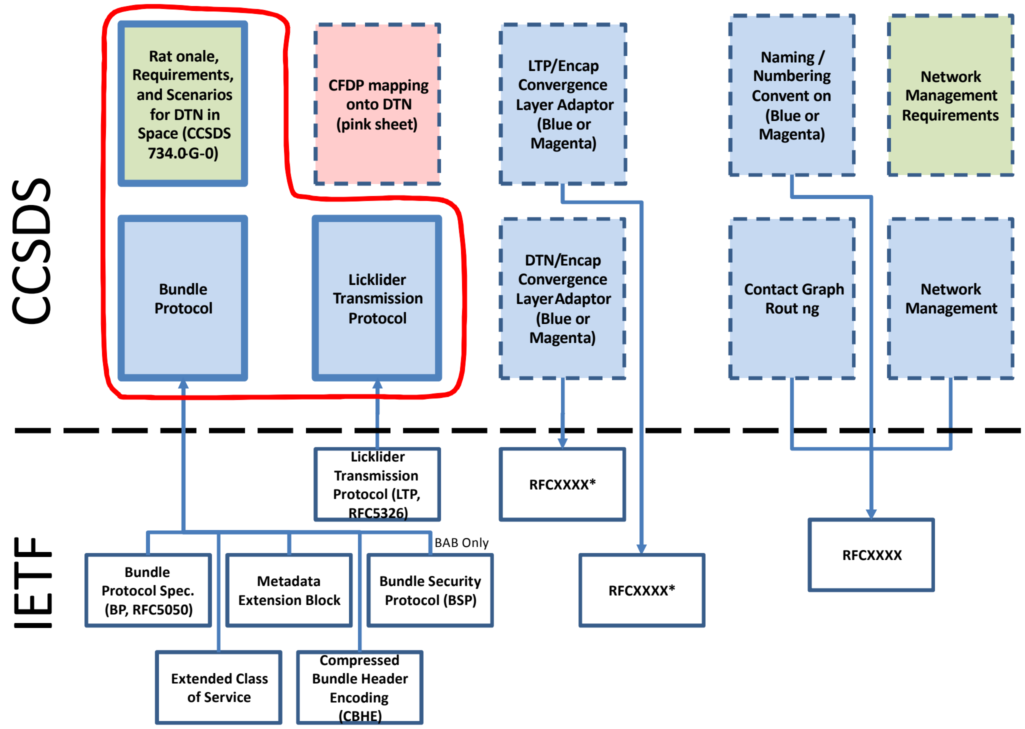

CCSDS
Rat onale, Requirements, and Scenarios for DTN in Space (CCSDS 734.0G-0)
CFDP mapping onto DTN (pink sheet)
LTP/Encap Convergence Layer Adaptor (Blue or Magenta)
Naming / Numbering Convent on (Blue or Magenta)
Network Management Requirements
Bundle Protocol
Licklider Transmission Protocol
DTN/Encap Convergence Layer Adaptor (Blue or Magenta)
Contact Graph Rout ng
Network Management
IETF
Licklider Transmission Protocol (LTP, RFC5326)
RFCXXXX*
RFCXXXX
BAB Only
Bundle Protocol Spec. (BP, RFC5050)
Metadata Extension Block
Bundle Security Protocol (BSP)
RFCXXXX*
Extended Class of Service
Compressed Bundle Header Encoding (CBHE)

**Figure 2-4: Document Map**

# 3 SCENARIOS

## 3.1 OVERVIEW

This section presents a number of scenarios that highlight issues to be considered in the definition of a DTN service.

## 3.2 SINGLE-SPACECRAFT WITH ONE GROUND STATION

The simplest scenario is a single control center communicating with a spacecraft via a single ground station, as shown in figure 3-1.

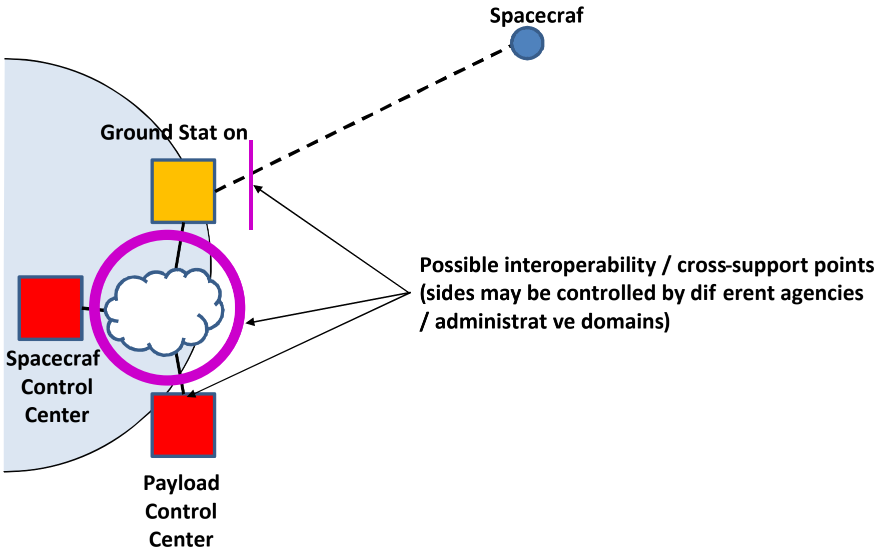

Spacecraf
Ground Stat on
Possible interoperability / cross-support points (sides may be controlled by dif erent agencies / administrat ve domains)
Spacecraf Control Center
Payload Control Center

**Figure 3-1: Scenario with a Payload Control Center, Spacecraft Control Center, Ground Station, and Spacecraft**

Issues to be considered in this scenario are:

- Connectivity to the spacecraft is usually predictable but may not be continuous:
  - The control center must know when the ground station is in communication with the spacecraft before it can begin transmitting.
- The link between the ground station and the spacecraft may be simplex.
- Contact periods between the ground station and the spacecraft may be of short duration (e.g., LEO satellites).
- The one-way light time (and hence the round-trip light time) between the ground station and the spacecraft may be large.
- Data from the spacecraft may need to be sent to the spacecraft control center, a payload control center, or both.

- The bandwidth into and out of the spacecraft control center may be such that forwarding all data for the payload control center through the spacecraft control center is not possible.
- The ground station and the spacecraft may unexpectedly lose connectivity.
- The spacecraft control center may need to ensure that data sent by the payload control center does not adversely affect the spacecraft.

## 3.3 COMMUNICATIONS WITH A LANDED ELEMENT VIA AN ORBITER THAT IS CONTROLLED SEPARATELY

In the scenario shown in figure 3-2, an orbiter control center is managing orbiter operations and a separate lander control center wishes to use the orbiter as a relay to communicate with a remote landed element. This situation generalizes to communicating with any node past the orbiter.

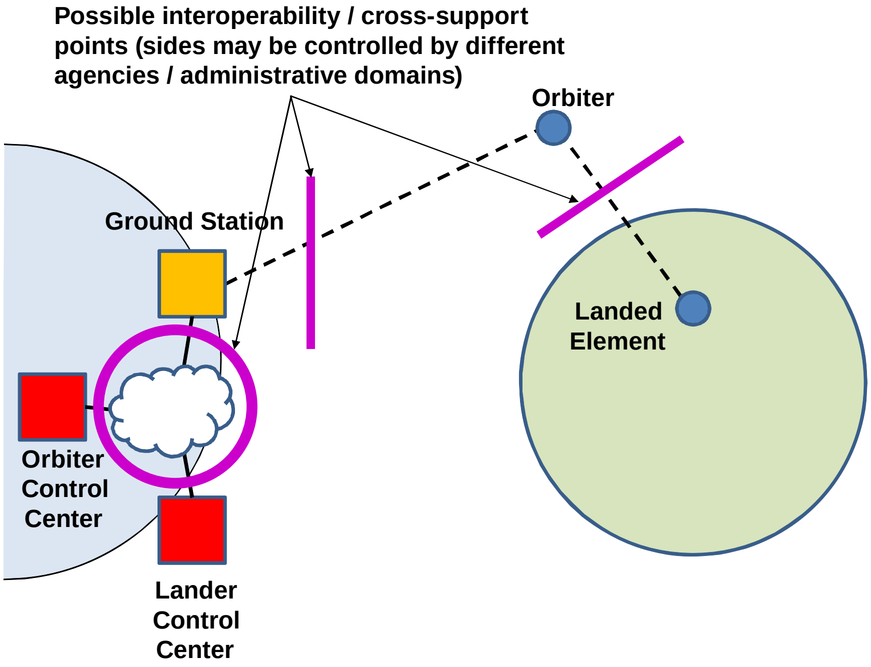

Possible interoperability / cross-support points (sides may be controlled by different agencies / administrative domains)
Orbiter
Ground Station
Landed Element
Orbiter Control Center
Lander Control Center

**Figure 3-2: A Relay Communications Scenario Where Different Assets May Be Controlled by Different Organizations/Agencies**

Issues to be considered in this scenario are:

- The orbiter control center may need to ensure that traffic for the lander does not have adverse affects on the orbiter:
  - traffic from the lander control center may need to be routed at the orbiter control center through an application that ensures no harmful commands are sent;

- the interface between the lander and orbiter control centers is a Network layer interface using the network protocol suggested by this document.

– Connectivity among the various elements may be intermittent.

– Connectivity among the elements may change in predictable or unpredictable ways.

– The orbiter may need to communicate with the lander in a low-level or 'emergency' mode, where not all of the networking capabilities of the lander are functioning. This is essential if the lander does not have a Direct-From-Earth (DFE) communication capability.

## 3.4 COMMUNICATIONS WITH A LANDED ELEMENT VIA ONE OR MORE ORBITERS THAT ARE CONTROLLED SEPARATELY

In the scenario shown in figure 3-3, an orbiter control center run by Agency A is in charge of some number of orbiters, and a separate lander control center wishes to use the orbiter services to communicate with a remote landed element. This situation generalizes to communicating with any node past the orbiter(s) controlled by Agency A.

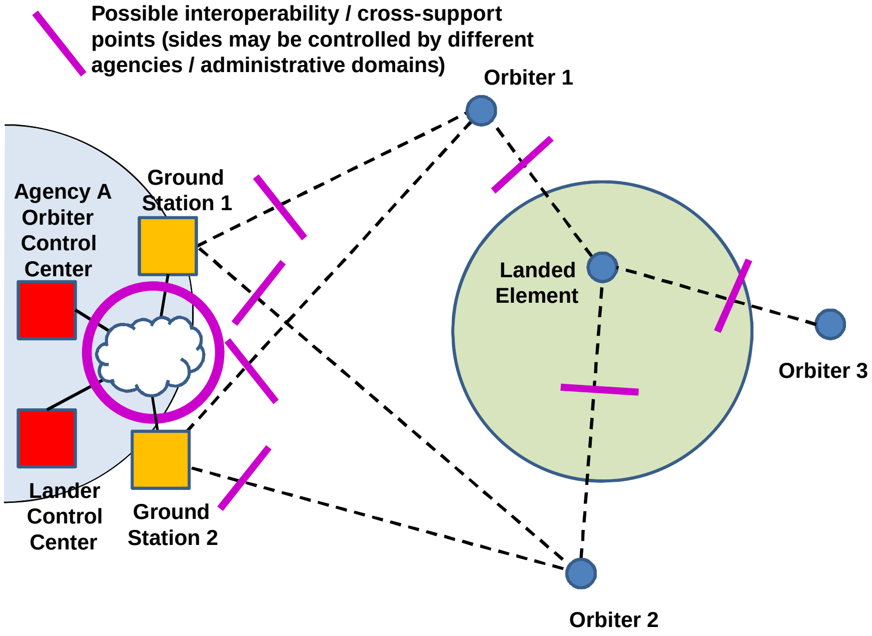

Possible interoperability / cross-support points (sides may be controlled by different agencies / administrative domains)
Orbiter 1
Agency A Orbiter Control Center
Ground Station 1
Landed Element
Orbiter 3
Lander Control Center
Ground Station 2
Orbiter 2

**Figure 3-3: A Relay Scenario Where Multiple Orbiters May Communicate with a Landed Element**

Issues to be considered in this scenario are:

– The data paths between the lander and the lander control center (which ground station(s) / orbiter(s) to use as relays) must be determined via some mechanism.

  - The orbiter control center(s) may need to ensure that traffic for the lander does not have adverse effects on the orbiter.
  - Connectivity among the various elements may be intermittent.
  - There may be times when it is better to wait for future connectivity rather than to transmit immediately since the lowest latency path to the destination may not be the one for which the first hop occurs soonest.
  - Connectivity among the elements may change in predictable or unpredictable ways.
- If two or more ground stations can receive from a spacecraft simultaneously, multiple copies of downlinked data may be received at the control center.
- For commanding, the control center may need to choose which ground station is to be used at a particular time. Conversely, the control center might want to send data to both ground stations and allow the one with the best connectivity to the spacecraft (as determined locally) to transmit.
- If both ground stations transmit simultaneously they may collide at the spacecraft receiver.
- The ground station with the best connectivity to the spacecraft may change in predictable or unpredictable ways.
- It may be best to transmit from one ground station and receive at another.

## 3.5 COMMUNICATIONS WITH A LANDED IN-SITU NETWORK

The scenario shown in figure 3-4 is the same as that in figure 3-3, except now there are multiple in-situ elements that use a local networking protocol to communicate amongst themselves.

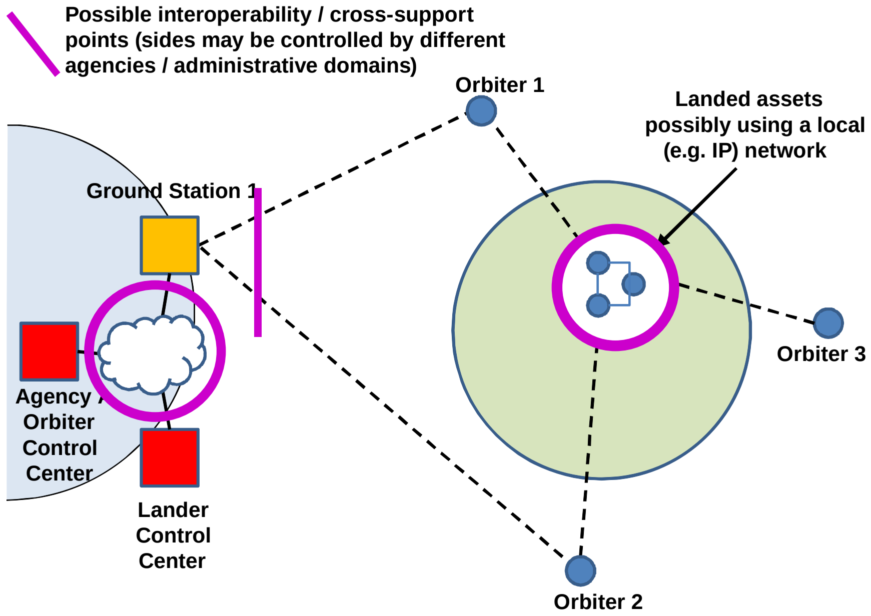

Possible interoperability / cross-support points (sides may be controlled by different agencies / administrative domains)
Orbiter 1
Landed assets possibly using a local (e.g. IP) network
Ground Station 1
Agency A Orbiter Control Center
Lander Control Center
Orbiter 3
Orbiter 2

**Figure 3-4: A Scenario Involving a Remote In-Situ Network That May or May Not Use the End-to-End Space Internetworking Protocols**

Issues to be considered in this scenario are:

– If the landed assets use a different internetworking protocol than that used for end-to-end communications, some sort of gatewaying or tunneling mechanism will need to be used.

# 4 REQUIREMENTS OF A SPACE INTERNETWORKING PROTOCOL

## 4.1 OVERVIEW

This section presents a set of system- or mission-level requirements derived in part from the scenarios and issues identified in section 3.

## 4.2 REQUIREMENTS OF A SPACE INTERNETWORKING SERVICE

### 4.2.1 SUMMARY OF NETWORK-LAYER REQUIREMENTS

Any space internetworking service must provide the following:

**An Optionally Reliable End-to-End Application Data Unit Delivery Service—**The space internetworking protocol must provide for the addressed, end-to-end delivery of octet-aligned, user-defined PDUs to application instances. It must be possible to de-multiplex PDUs to a number of different upper-layer services, including specific applications (i.e., it must be possible to address a PDU to a specific application on board the spacecraft, not just to the spacecraft as a whole). This service *to applications* may be provided over a number of underlying services, including space data links or existing network and transport services such as provided by the Internet protocols.

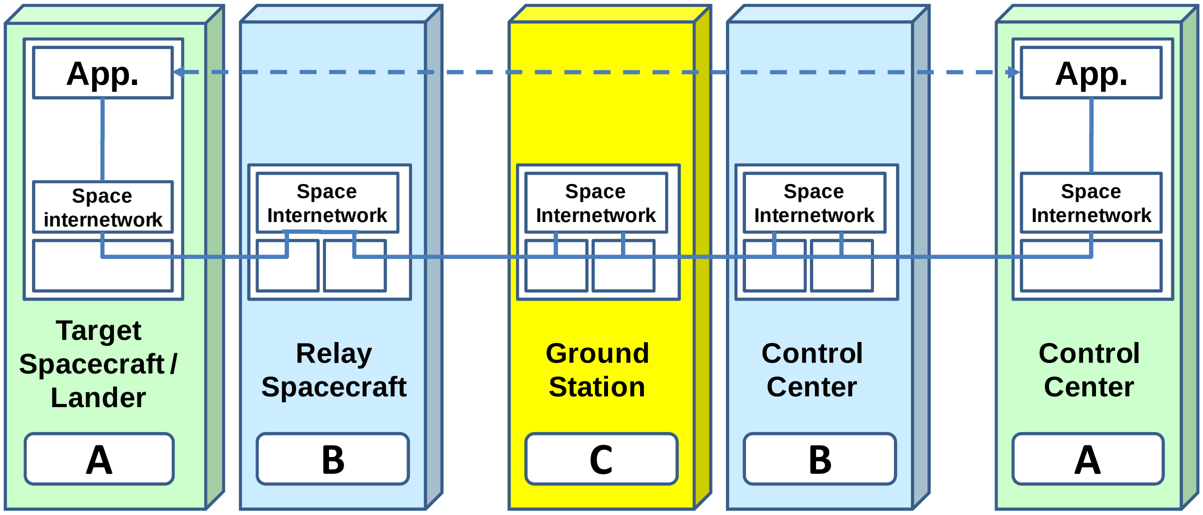


**Figure 4-1: Basic Space Internetworking Service**

In the figure, the different letters underneath the various assets represent ownership by different agencies. An important goal of space internetworking is to allow the different elements in figure 4-1 to be operated by different agencies while allowing automated, end-to-end data flows. Standardizing the space internetworking protocol and the relevant lower layers in the communication stack will enable the cross support necessary to allow these flows.

The Space Internetworking effort does not address the operation of one agency's spacecraft by another agency. While standardizing on a common data transport mechanism enables this, the protocol mechanisms for application interoperability are beyond the scope of this document.

**Ability to Handle Arbitrarily Sized Application Layer PDUs—**The size of the Application layer data units transported by the space internetworking protocol should not be constrained by the underlying technologies used.

**End-to-End SDU Delivery in the Presence of Delays/Disruptions—**For space communication there may be multiple sources of delay and disruption, some planned and some not. Planned delays include light-time latencies that range from minutes to hours and beyond for deep-space communication. Disruptions include planned resource scheduling issues that restrict connectivity to certain windows, unplanned reallocation of resources that may interrupt communications, and unplanned disruptions due to environmental factors (e.g., multipath, solar activity). Thus any space internetworking protocol must be able to function in the presence of the following environmental constraints:

– ***long delays***—when even the data link Round Trip Time (RTT) may be measured in minutes or hours;

– ***temporary network partitioning***—when there is no network path to the destination for some period of time;

– ***half-duplex communication paths***—when communication is one-way for some period of time:

    - it is expected that if A can send information to B then there will be some time in the future when B can send information to A, but that it is possible that any such reverse path may not be available at the same time as the forward path;

    - individual links may be completely simplex;

– ***contacts that do not support entire Network layer PDUs*—**when no single contact contains enough bandwidth to forward an entire Network layer PDU; in these cases, the space internetworking protocol must be capable of fragmenting the Network layer PDU so that it can be forwarded in pieces and reassembled before presenting it to the next higher layer.

**Data Accountability—**A space internetworking protocol must provide mechanisms to ascertain where particular network PDUs are in the network and, if they have been discarded, the reasons for discarding them.

**Optional Reliability—**Both reliable and unreliable data delivery mechanisms are needed for space communications. Some data such as low-priority cyclic telemetry is well served by an unreliable delivery model. If a particular piece of data is lost, it is better simply to wait for the next sample than to waste resources retransmitting old (and presumably less useful) data. Alternately file transfers, messaging, and other applications will benefit from a common, standardized service that provides reliable data delivery. Providing reliable delivery as a network service frees applications from having to implement reliability and improves interoperability among applications needing reliability.

**Prioritized Data Delivery—**Not all data should be treated equally by the network service. More important data should have a higher probability of being delivered, be delivered sooner, or both. A space internetworking service needs to provide mechanisms for applications to signal the importance of data and needs to provide 'better' service to the more important data. To allow higher-priority data to be delivered sooner, mechanisms within the network should, as configured by policy and network management, allow higher-priority data to be transmitted before lower-priority data when both are queued for transmission from a node.

**Data Link Layer Agility—**The space internetworking protocol must be able to function over a wide range of Data Link layers, including at least CCSDS AOS, TC/TM, and Proximity-1 (references [2]-[5]). The space internetworking protocol must be able to function over paths composed of heterogeneous data links.

**Compatibility with the Terrestrial Internet—**The space networking protocol must be transportable across the terrestrial Internet.

**Security—**The space internetworking protocol SHOULD provide security mechanisms for:

– authenticated access to the network;

– integrity and confidentiality services for user data.

Whether or not the various security mechanisms are invoked MUST be controllable by policy.

**Management—**The space internetworking protocol SHOULD provide mechanisms for:

– network management;

– automated route construction and maintenance (dynamic routing).

**Support for Higher-Layer Services—**It must be possible to construct at least the following higher-layer services from the internetworking service:

– file delivery service—a service to deliver files (bounded groups of octets) together with metadata (e.g., file names, file locations) to remote file stores;

– messaging service—a service to deliver bounded groups of octets together with metadata to applications;

– Space Packet delivery service—a service to move Space Packets across a multi-hop infrastructure and to deliver them to specific applications.

### 4.2.2 SYSTEM REQUIREMENTS

#### 4.2.2.1 General

This subsection provides a detailed list of *system-level* requirements derived from an ESA study on file-based mission operations. This set of requirements is deemed sufficient to be able to execute space missions, and is provided here to provide context and rationale for the requirements on the internetworking protocols being described in this document.

Figure 4-2 shows the overall communications stack including two main classes of applications. The 'service applications' are direct users of the space internetworking protocol and perform common functions such as file transfers and messaging. It is expected that the service applications will be invoked by other applications to carry out spacecraft functions, while the 'user applications' will be the users of the service applications. User applications may themselves access the space internetworking services directly.

The space internetworking protocol runs on top of data link protocols at each link. The data link protocols may provide services that include, for example, reliable data delivery, resilience against temporary link disruptions, and segmentation/reassembly of internetworking PDUs to adapt them to the characteristics of the link.

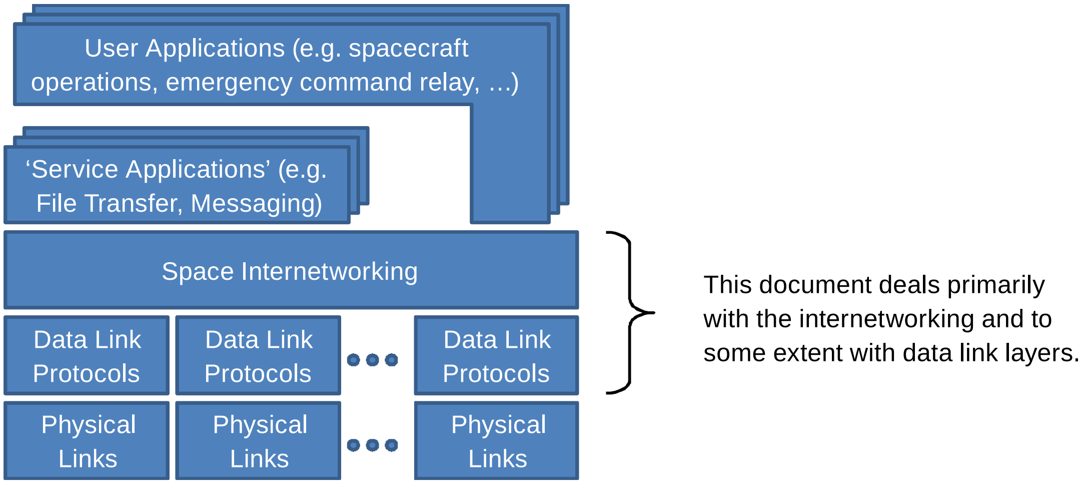

User Applications (e.g. spacecraft operations, emergency command relay, …)
'Service Applications' (e.g. File Transfer, Messaging)
Space Internetworking
Data Link Protocols
Data Link Protocols
Data Link Protocols
Physical Links
Physical Links
Physical Links
This document deals primarily with the internetworking and to some extent with data link layers.

**Figure 4-2: A Space Internetworking Architecture Protocol Stack**

NOTE – The above figure is notional; it is not in the scope of this document to define an overall architecture for CCSDS missions or space internetworking. The requirements here may provide useful input to future working groups seeking to perform such tasks.

#### 4.2.2.2 General Requirements

##### 4.2.2.2.1 Communications shall be supported to a spacecraft via zero or more intermediate relays.

Rationale: The intent of the internetworking layer is to support multi-hop data transfers in an automated fashion.

NOTE – Communication with zero relays is equivalent to direct Data Link layer communication between the endpoints.

NOTE – Examples of entities that may perform relay functions include: ground facilities, Earth stations, data relay satellites, landers, and internal spacecraft nodes.

##### 4.2.2.2.2 It shall be possible to use local, in-situ networking technologies different from the end-to-end space internetwork technology.

Rationale: It may be desirable to use different technologies at different points along a multi-hop path, each tuned to the local environment to improve performance.

##### 4.2.2.2.3 The system shall support a general class of applications, including at least file transfer and messaging.

Rationale: The point of an internetwork is to support application layer communications.

NOTE – The currently envisioned applications include file transfer and messaging as might be implemented via the CFDP, AMS, and Message Abstraction Layer (MAL) [6] protocols. The system may also need to support the transfer of TM/TC packets *over* the internetwork protocol (tunneling TM/TC over the internetwork protocol).

NOTE – The specification of PDUs for MAL transport over a given technology are contained in technology binding specifications. Such a technology binding for DTN-BP did not exist at the time of writing, but as DTN-BP is intended to be a general purpose networking protocol, no barrier to the specification of such a technology binding has been identified.

##### 4.2.2.2.4 Management information relating to data transfer shall be collected in all nodes.

Rationale: Management information will be required to operate the network.

NOTE – ‘Network operators’ may include people and/or automated programs.

**4.2.2.2.5 Management information relating to data transfer shall be made available to network operators.**

Rationale: Management information will be required to operate the network. Information is useful only if it is available to the right people/places.

**4.2.2.2.6 Network operators shall be able to manipulate management information in all nodes.**

Rationale: Network operators will be able to use this function to control operations of network nodes.

NOTE – It is assumed that this requirement will be fulfilled by a network management function distinct from the network protocols themselves.

**4.2.2.2.7 It shall be possible to configure routing to automatically fail over to redundant routes if such routes are available.**

Rationale: Operators may want to configure automated failover paths so that if there is a problem forwarding data along the primary path, data are re-routed along backup paths without having to wait for operator intervention.

**4.2.2.2.8 Communications firewalls shall be implemented at interoperability points to guarantee mission security.**

Rationale: Network-layer attacks, especially on networks connected to the Internet, are common. Firewalls will help protect flight networks from these attacks.

NOTE – This requirement refers to IP-layer firewalls needed for infrastructure security on the ground and is not a requirement on the space internetworking protocols.

**4.2.2.2.9 Methods for user authentication shall be incorporated with authenticated users having associated levels of permission and resource allocation.**

Rationale: User authentication and resource control will help protect assets from over-allocation (accidental or malicious) by users.

NOTE – This requirement may apply at multiple levels of the stack. At the Application layer it may apply to users’ ability to consume space in the file system(s) of end nodes; at the Network layer it implies that the Network layer will be able to accept some sort of authentication token and be able to use that token to make decisions about resource usage/ allocation. Further, the Network layer shall be able to carry the token as part of the Network layer PDU and pass the token to lower layers of the communication stack as their service interfaces permit.

**4.2.2.2.10 Data privacy between users shall be provided.**

Rationale: Users may want to prevent other users from accessing their data. For example, private medical data for astronauts may need to be protected.

NOTE – This is primarily a requirement on end systems and applications. In particular it does NOT imply that the internetwork layer must provide cryptographic protection of user data.

**4.2.2.2.11 It shall be possible to use multiple ground stations to communicate with a single space asset with some ground stations providing downlink capability only.**

Rationale: It is expected that the connectivity pattern in the requirement will be relatively common.

NOTE – The TDRSS system, for example, provides an on-demand downlink-only capability.

**4.2.2.2.12 It shall be possible to route data from the ground station directly to destinations without routing via the control center.**

Rationale: It could be advantageous to be able to bypass the control center for downlinked data in the case that capacity of the ground station to control center network is too low to send all the data via the control center within an acceptable time frame.

**4.2.2.2.13 It shall be possible to implement application layer firewalls at interoperability points to guarantee mission safety.**

Rationale: It is expected that some asset managers will require this capability.

NOTE – For example, an orbiter operation center may require that all traffic to a lander that goes through the orbiter pass through the orbiter control center so that it can be examined. The mechanism to force such traffic through the correct firewall nodes is achieved by appropriate formation of the routing tables.

**4.2.2.2.14 ‘Low-level commanding’ of spacecraft by embedding special command sequences in data link layer frames shall be supported.**

Rationale: Spacecraft often require mechanisms to respond to very low-level commands in case higher spacecraft functions are not available or are not operating correctly.

NOTE – Mechanisms for hardware commanding of remote (several network hops away) spacecraft are described in more detail in 4.2.3 below.

#### 4.2.2.3 Data Transport Requirements

**4.2.2.3.1 It shall be possible to send a file to a spacecraft application that can, either by autonomous methods or managed by mission/infrastructure management or a combination of both, convey the file to a second spacecraft.**

Rationale: It is expected that some mission operators will desire this capability.

NOTE – The Network layer protocol will support a generic class of applications, one of which could certainly implement this functionality. This capability is not meant to replace the functions of the Network layer.

**4.2.2.3.2 The end-to-end infrastructure and protocols shall be capable of transferring, as SDUs, the PDUs of the following CCSDS protocols: CFDP, Space Packet Protocol (SPP), Encapsulation Packet Protocol (EP), Telemetry (TM), Telecommand (TC), Message Abstraction Layer (MAL), and Asynchronous Messaging System (AMS).**

Rationale: It is expected that the above-listed services will form the basis for spacecraft operations; the internetworking layer needs to support them.

NOTE – This presumes some sort of application endpoint to consume, e.g., raw Space Packets.

NOTE – The specification of PDUs for MAL transport over a given technology are contained in technology binding specifications. Such a technology binding for DTN-BP did not exist at the time of writing, but as DTN-BP is intended to be a general purpose networking protocol, no barrier to the specification of such a technology binding has been identified.

**4.2.2.3.3 The end-to-end infrastructure and protocols shall provide the services specified as required of the underlying layers of the CFDP, SPP, EP, Telemetry, Telecommand, MAL, and AMS protocols.**

Rationale: Because the space internetworking protocol may carry the above-listed packet formats, it must be able to provide the services required of those protocols.

NOTE – A shim above the internetworking protocol itself (similar to the UT layer in CFDP) may be used to support these or other higher-layer protocols.

**4.2.2.3.4 The end-to-end infrastructure and protocols shall be capable, under the direction of users and/or mission/infrastructure network management, of supporting qualities of service with respect to data completeness.**

Rationale: Some applications will require complete data delivery while others will not.

**4.2.2.3.5 The end-to-end infrastructure and protocols shall be capable, under the direction of users and/or mission/infrastructure network management, of supporting qualities of service with respect to data errors.**

Rationale: Some applications will require error-free data delivery while others will not.

**4.2.2.3.6 The end-to-end infrastructure and protocols shall be capable, under the direction of mission/infrastructure network management, of supporting qualities of service with respect to data sequencing (depends on tolerance to out-of-sequence PDUs of upper layer protocols).**

Rationale: Some applications require in-sequence delivery of multiple Application layer data units.

NOTE – The sequence-preservation mechanism may be provided by Network layer mechanisms or by some protocol between the application and Network layers.

**4.2.2.3.7 The end-to-end infrastructure and protocols shall be capable, under the direction of the application and mission/infrastructure network management, of supporting Quality of Service (QoS) with respect to data priority.**

Rationale: Not all data require the same QoS with respect to priority.

NOTE – This requirement ensures that applications, under the guidance and control of policy, can mark which data they think is more 'important' than other data.

**4.2.2.3.8 The end-to-end infrastructure and protocols shall be capable, under the direction of users and/or mission/infrastructure network management, of supporting qualities of service with respect to data availability (via, for example, alternate routes).**

Rationale: Some data are so important that they need to be forwarded along multiple parallel paths to increase probability of delivery and/or to reduce end-to-end latency.

**4.2.2.3.9 The Space Internetworking Protocols (e.g., BP and IP) shall be capable of operating over the CCSDS Encapsulation Protocol.**

Rationale: The Encapsulation Packet provides a way of multiplexing the internetworking protocols with other CCSDS packet types on CCSDS links.

#### 4.2.2.4 Data Transfer Requirements

**4.2.2.4.1 The transfer protocols shall be capable of transferring application data units completely (reliably) when required by applications. If an application does not require complete delivery, the transfer protocols may deliver incomplete data (data with holes).**

Rationale: Some applications may require complete data delivery while others may not.

**4.2.2.4.2 The transfer protocols shall be capable of transferring complete sequences of messages.**

Rationale: Applications may need to transfer complete sequences of messages.

**4.2.2.4.3 The transfer protocols shall be capable of transferring sequences of messages in sequence.**

Rationale: Some applications will want to send sequences of messages and have them be delivered to the destination in sequence.

NOTE – The sequence-preservation mechanism may be provided by Network layer mechanisms or by some protocol between the application and Network layers.

**4.2.2.4.4 It shall be possible to transfer a file over a disrupted link, retaining the state of the file transfer between contact periods.**

Rationale: Some file transfers may have to be broken across contacts.

**4.2.2.4.5 It shall be possible to 'hand-over' the transmission of a file from one intermediate hop to another (e.g., transmission starts using ground station A, A looses visibility and hands-over to ground station B).**

Rationale: Some file transfers may have to be handed over between next-hop nodes.

**4.2.2.4.6 File and message transfer shall be capable of operating over simplex links (with limited QoS).**

Rationale: Some space links may be simplex; the internetworking protocols need to function in these environments but may not be able to provide all of the QoS services (such as reliability).

**4.2.2.4.7 File and message transfer shall be capable of operating over network paths with widely differing capacities (up to 10,000:1)**

Rationale: Some space links will have high degrees of asymmetry in data rate.

NOTE – The asymmetry constraint is probably a constraint on all applications. It should be noted that 'operate' pertains to functionality, not efficiency, and that while this requirement imposes constraints on the network protocol(s), it cannot be met by the network protocol(s) alone: the file and message protocols themselves affect Rationale: the ability to operate over asymmetric paths.

**4.2.2.4.8 File and message transfer protocols shall be independent of file and message contents.**

Rationale: The file and message transfer protocols should not need to be cognizant of the data content of the files/messages being transferred. This is a requirement on specific applications.

**4.2.2.4.9 File transfer may be initiated by the sender of a file, the receiver of a file, or a third party.**

NOTE – This is taken directly from CFDP and is a requirement on a file transfer application. It has nothing to do with internetworking protocols.

**4.2.2.4.10 File transfer shall take place between file stores under the control of file service user entities.**

NOTE – This is taken directly from CFDP and is a requirement on a file transfer application. It has nothing to do with internetworking protocols.

**4.2.2.4.11 Message transfer shall take place between message service user entities.**

**4.2.2.4.12 Data transfer shall be possible over multiple concatenated heterogeneous data transport layers.**

Rationale: The intent of an internetworking layer is to allow transport across multiple underlying data links.

NOTE – Here 'data transport layers' refers to whatever technology is used by the internetworking service, including possibly OSI layer-4 services (e.g., TCP).

**4.2.2.4.13 Given suitable QoS attributes when data is submitted and suitable network connectivity, it shall be possible to verify completeness of the data transfer and to notify the data transfer originator about this. This shall be possible regardless of other QoS attributes (e.g., completeness).**

Rationale: Data senders may want to know if data has been completely transferred.

NOTE – The preconditions above are meant to ensure that such notification is desired and possible. For example, if there is no return path from the receiver to the sender it will not be possible to notify the sender of anything.

**4.2.2.4.14 Data transfer shall support priority and preemption mechanisms in all nodes.**

Rationale: Priority and preemption may be required to ensure that important data are delivered in a timely manner.

NOTE – It is understood that preemption may not be possible at all layers of the stack. A radio might consume a buffer of data and not allow preemption until that entire buffer has been sent, for example.

**4.2.2.4.15 It shall be possible to transfer file metadata as part of the file transfer protocol or using a messaging protocol.**

NOTE – Applications should have flexibility in the way they are designed and implemented. This is an Application layer requirement and has no impact on the internetworking protocols.

**4.2.2.4.16 Data transfer protocols shall not require simultaneous availability of the communication link between all nodes involved in the data delivery/routing.**

Rationale: Contemporaneous end-to-end paths are not guaranteed to exist in the space internetworking environments in which the proposed internetworking protocol needs to operate.

**4.2.2.4.17 It shall be possible to use the same data transfer protocol in the ground-to-space link, in the space-to-space link, and between ground nodes (ground-to-ground).**

Rationale: A single data transfer protocol that operates over a series of possibly heterogeneous underlying layers provides a single service to applications above it, allowing the applications to be simpler and to concentrate on Application layer operations rather than on interfacing to the lower layers.

NOTE – This does not preclude using some other data transfer mechanism for local, in-situ communications.

**4.2.2.4.18 Data retransmission strategy shall be flexible to allow opportunistic (automated) retransmission of data when links become available while still respecting QoS conditions.**

Rationale: Sometimes data will be lost and will need to be retransmitted; automated retransmission will improve overall performance.

**4.2.2.4.19 Retransmitted data shall, by default, assume the same priority as the original data.**

Rationale: Retransmitted data must by default have some priority, and using the priority of the original transmission makes sense.

**4.2.2.4.20 The priority and queue position for retransmitted data may be modified by policy, or by local or remote network management.**

Rationale: It may be desirable to override the default parameters for retransmitted data by policy or network management.

**4.2.2.4.21 It shall be possible to demultiplex the SDUs contained in Network layer PDUs to specific upper-layer entities.**

Rationale: It is desirable for multiple applications to be able to share the internetworking service. Multiplexing and demultiplexing SDUs enables this.

**4.2.2.4.22 The data transfer protocols shall be able to operate in a communications environment characterized by large transmission delays.**

Rationale: Some space links will have large transmission delays.

**4.2.2.4.23 The data transfer protocols shall be able to operate in a communications environment characterized by unreliable, noisy communication links.**

Rationale: Some space links will have moderate to high bit error rates (be ‘noisy’).

**4.2.2.4.24 The data transfer protocols shall be able to operate in a communications environment characterized by interrupted visibility between communication nodes due to predictable causes (e.g., orbital visibility).**

Rationale: Connectivity between nodes of the space internetwork will often be limited by predictable circumstances such as visibility and operations constraints.

**4.2.2.4.25 The data transfer protocols shall be able to operate in a communications environment characterized by unpredictable disruptions due to failures.**

Rationale: Unplanned disruptions and failures will sometimes occur, and the data transfer protocols need to not fail because of them.

**4.2.2.4.26 The protocol shall have a mechanism for carrying a priority field that may be affected by the user and/or management/policy at the sending node.**

Rationale: Users may have some data that they think is more important than other data. The system may shape the priorities assigned to user traffic.

**4.2.2.4.27 Management/policy at intermediate nodes (nodes other than the source) may override the priority treatment indicated in the priority field of a space internetworking PDU.**

Rationale: The system may need to shape the priorities assigned to user traffic under the control of network management in order to provide acceptable quality of service to all users.

**4.2.2.4.28 It shall be possible for the file transfer protocol to perform multiple file transfer transactions in parallel (e.g., in order to initiate the delivery of file 'n+1' before receiving confirmation of successful transfer of file 'n'). This is essential in order to optimize the use of the available bandwidth.**

Rationale: Allowing multiple concurrent file transfers will improve efficiency by allowing the file transfer protocol to continue transmitting even if it has not received an acknowledgement for previously sent files.

NOTE – The desire here is to have a file transfer protocol that can initiate multiple concurrent transfers in an effort to keep the 'pipe' (the path between source and destination) full of data. If just a single file transfer were allowed at any given time, then it might be that there would be times when a node that could be transmitting would be idle while waiting for, e.g., an acknowledgement. This is a requirement on a file transfer application that is beyond the scope of this WG. It has no impact on the networking protocols.

### 4.2.2.5 Data Management Requirements

**4.2.2.5.1 It shall be possible to observe the progress of data transfers by local or remote data management entities.**

Rationale: It is expected that mission operators will desire the ability to be able to observe the progress of data transfers.

NOTE – This capability can be provided by a combination of the data transfer application and a network capability to locate individual network PDUs.

**4.2.2.5.2 It shall be possible to observe the state of data transfer queues (file or message) by local or remote data management entities.**

Rationale: It is expected that mission operators will desire the ability to observe the state of data transfer queues.

NOTE – This capability can be provided a combination of the file/message transfer application and a network capability to locate individual network PDUs.

**4.2.2.5.3 It shall be possible to control data transfer queues by reordering, deleting, suspending/resuming transmission of queued items by local or remote data management entities.**

Rationale: It might be advantageous for network management entities to re-order or delete queued items in response to unforeseen circumstances.

#### 4.2.2.5.4 It shall be possible to control the actions of file transfer applications with respect to stop (cancel), suspend, and resume (global or individual files) by local or remote data management entities.

Rationale: It is expected that mission operators will desire the ability to perform the above-listed operations on file transfers.

NOTE – The above-listed capabilities can be provided a combination of the file transfer application and a network capability to locate and manage (delete, suspend, resume) individual network PDUs.

#### 4.2.2.5.5 It shall be possible to preempt data transfers either locally to the sending entity or remotely from a remote manager.

Rationale: It is expected that mission operators will desire the ability to preempt data transfers to allow higher priority data to flow more quickly through the network.

NOTE – This capability can be provided by a combination of the data transfer application and a network capability to locate and manage (delete, suspend, resume) individual network PDUs.

#### 4.2.2.5.6 Suspension and resumption of transfer at transmitting or receiving ends may be initiated by a local management entity in response to an anticipated or unanticipated outage.

Rationale: It may be advantageous to be able to ‘freeze’ transmission in response to outages.

#### 4.2.2.5.7 It shall be possible to establish primary and backup routes through the end-to-end data path at a network planning facility and to distribute this information to the nodes concerned.

Rationale: By establishing and distributing backup route information, the system can react locally to unplanned changes without having to wait for operator intervention.

#### 4.2.2.5.8 Synchronization of route changes must be managed in the end-to-end network.

Rationale: If changes are to happen, it may be advantageous or required to effect those changes at multiple nodes simultaneously.

#### 4.2.2.5.9 It shall be possible to terminate data transmission via a relay node *A*, delete the data buffered at *A*, and resume data transmission via another next-hop relay, if necessary.

Rationale: If a more desirable transmission opportunity becomes available, it may be desirable to terminate an ongoing transmission and switch to the more desirable one.

**4.2.2.5.10 The data transfer protocols shall provide to the destination the time of transmission and receipt of the application data unit being delivered.**

Rationale: Applications may want to know these times.

#### 4.2.2.6 Data Utilization Requirements

**4.2.2.6.1 Application layer content (e.g., files, messages) for onward transmission to a spacecraft may be examined and checked for mission critical effects at a mission control entity and blocked if necessary.**

Rationale: Mission operators may want to inspect traffic flowing through their mission assets to ensure the assets' safety.

NOTE – The routing mechanisms needed to ensure that the correct data gets to the 'checker' are a function of building the routing tables; the checking mechanism is an Application layer function.

**4.2.2.6.2 In the case that a mission control entity blocks some Application layer content from being forwarded to a spacecraft, the entity shall notify the sender that the data was blocked.**

Rationale: The data sender should know that it has misbehaved so that it can modify its behavior.

**4.2.2.6.3 An application on the last hop relay node may extract TCs from an immediate or delayed TC file and radiate them as TCs to their destination (typically orbiter to lander).**

Rationale: Such an application could be used to support low-level commanding in case the destination spacecraft's network layer is not functioning properly.

**4.2.2.6.4 An application on the first hop relay node may assemble TM packets received from another entity and assemble them into a TM file for further transmission.**

Rationale: Such an application could be used to extract data from a spacecraft whose network layer is not functioning properly.

### 4.2.3 LOW-LEVEL COMMANDING AND TELEMETRY

Spacecraft often require mechanisms to respond to very low-level commands in case higher spacecraft functions are not available or are not operating correctly. Such commands are typically used to reboot the C&DH or to place the spacecraft into a known state, usually in preparation for re-starting higher layer services. A peer ability to receive low-level telemetry from a spacecraft whose networking services are not functioning is similarly required. This subsection describes how such services can be implemented leveraging the multi-hop communications of the network service.

An orbiter, for instance, thus performs a basic control and monitoring service to the lander using low-level commands and telemetry formats which directly access the point-to-point link between orbiter and lander. These low-level commands can put the lander into a known state, even when higher layer (e.g., networking) functions are not available due to single-event-upsets or other equipment failure.

Thus the steps in the low-level commanding process could be:

a) Low-Level Commanding (LLC) application generates the low-level command(s) for the target spacecraft.

b) The LLC application sends the commanding data addressed to the LLC proxy application at the proximate relay, most likely using a file transfer protocol. Together with the low-level command(s), the sending LLC application identifies the target spacecraft and any extra parameters needed to issue the low-level commands.

c) The LLC message is forwarded to the Proximate Relay via the network layer. This requires that all other network-layer relays in the path are functioning properly.

d) The LLC application on the proximate relay consumes the LLC application data and generates the data link layer frame(s) containing the commands to transmit to the target spacecraft.

e) The frame(s) containing the LLC commands are sent to the target spacecraft. It is assumed that these commands can be processed when neither the networking or higher-layer services on the target spacecraft are available.

f) The LLC is detected and acted on by the target spacecraft.

Conversely, low-level telemetry functions can be used by elements that are unable to send networked data. Using a point-to-point link between the end node and a relay and low-level telemetry, an end node can transfer data to an application on the relay which can then use higher-layer functions such as file transfer to forward the data to Earth.

The steps in the low-level telemetry process could be:

a) End node sends data to relay application using Low-Level Telemetry (LLTM). The mechanisms to set up this transfer are outside the scope of what is described here, but could result from low-level commanding of the end node, e.g.

b) LLTM application on the relay consumes the data and uses higher-layer services such as a file transfer service to send the data to the mission control center or other destination. This transfer requires that

c) The LLC message is forwarded to the Proximate Relay via the network layer. This requires that all other network-layer relays in the path are functioning properly.

d) The LLC application on the proximate relay consumes the LLC application data and generates the data link layer frame(s) containing the commands to transmit to the target spacecraft.

e) The frame(s) containing the LLC commands are sent to the target spacecraft. It is assumed that these commands can be processed when neither the networking or higher-layer services on the target spacecraft are available.

f) The LLC is detected and acted on by the target spacecraft.

### 4.2.4 SUPPORT FOR HIGHER-LAYER SERVICES

File transfer is an application that is increasingly gaining acceptance in the space community, especially for the delivery of telemetry. Using the file transfer model, 'files' of telemetry information are created on board the spacecraft for transmission to the ground. The CCSDS File Delivery Protocol (CFDP) was designed to meet this need in environments where the source and destination are connected by a single data link. CFDP's extended procedures and store-and-forward overlay procedures were designed to address multi-hop communications paths, but lack the power and flexibility of the Bundle Protocol to deal with multiple, possibly changing, multi-hop network paths such as from a lander on Mars via one of two or more orbiters to Earth.

While a new file transfer service could be built on top of the space internetworking service, it makes more sense (at least in the near term) to retain CFDP to provide the file transfer service and to use the space internetworking service as the Unitdata Transfer (UT) service of CFDP. This will allow existing applications and software that use the CFDP interface to continue without modification while providing enhancements from the Bundle Protocol such as multi-hop routing. Operating CFDP over an internetworking service will immediately allow CFDP to address scenarios 4 and 5 (reference [8]): multi-hop file delivery where parts of the file are sent reliably along different paths to the destination.

DTN can easily implement CFDP Scenario 4 (reliable/unreliable end-to-end transfer via multiple waypoints in parallel) as shown in figure 4-3. CFDP segments, encapsulated in DTN PDUs, can be forwarded over multiple paths to the destination. This is trivially extended to the case where there are multiple serial hops along one or the other of the paths.

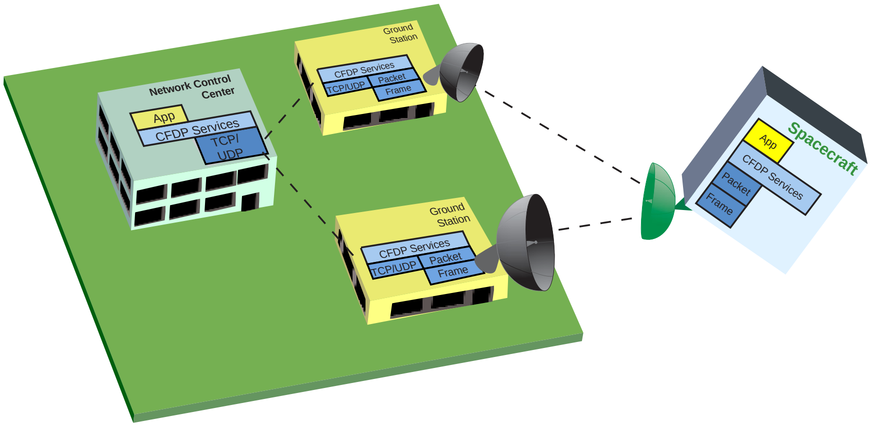


**Figure 4-3: CFDP Scenario 4**

Figure 4-4 shows how CFDP could be migrated to use a DTN service, including an intermediate stage that allows a CFDP implementation to communicate with both ‘old’ (non-DTN) and ‘new’ (DTN-based) implementations. This makes use of the layering internal to most CFDP implementations at the Underlying Transport Adaptor layer. Using this approach, CFDP implementations migrate from the configuration on the left to the one on the right. The part in the dotted oval on the right represents the ‘forward migration’ of the old architecture.

This represents a completely seamless growth path for CFDP from the current implementation to one based on DTN.

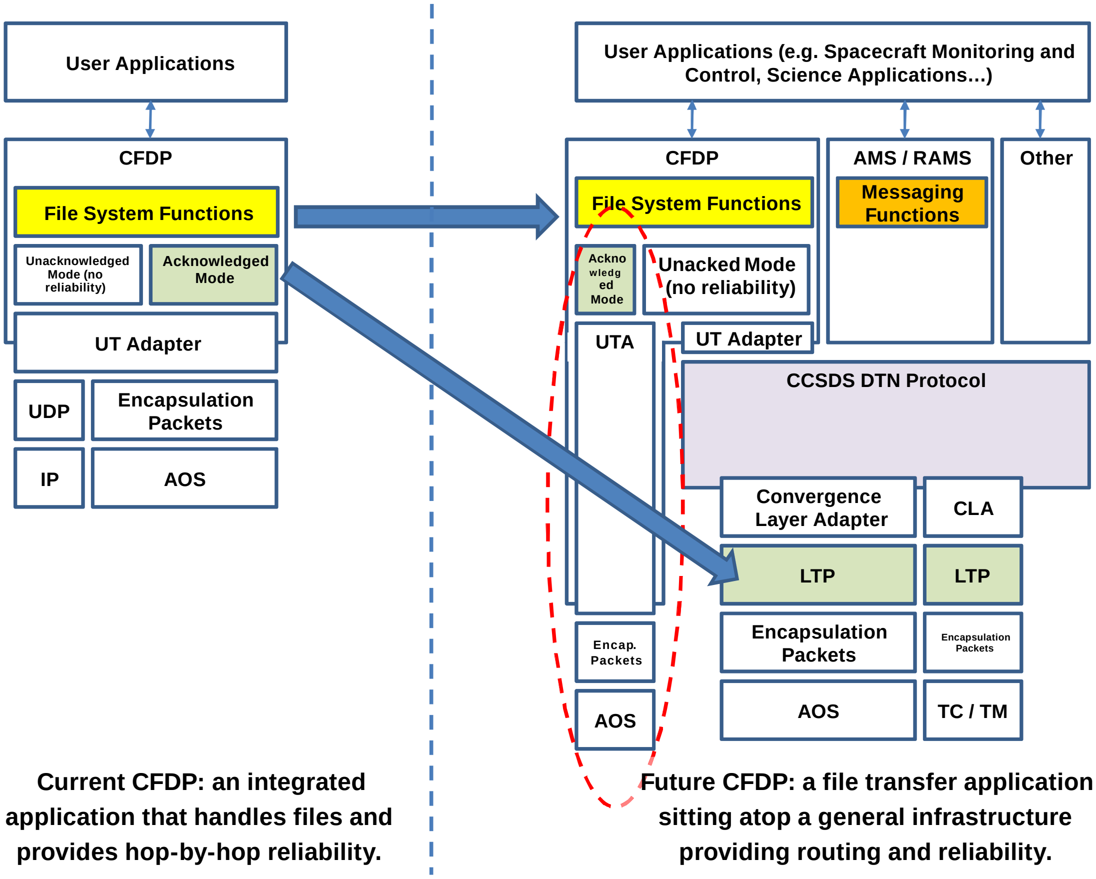

User Applications
CFDP
File System Functions
Unacknowledged Mode (no reliability)
Acknowledged Mode
UT Adapter
UDP
Encapsulation Packets
IP
AOS
User Applications (e.g. Spacecraft Monitoring and Control, Science Applications...)
CFDP
File System Functions
Ackno wledg ed Mode
Unacked Mode (no reliability)
AMS / RAMS
Messaging Functions
Other
UTA
UT Adapter
CCSDS DTN Protocol
Convergence Layer Adapter
CLA
LTP
LTP
Encap. Packets
Encapsulation Packets
Encapsulation Packets
AOS
AOS
TC / TM
Current CFDP: an integrated application that handles files and provides hop-by-hop reliability.
Future CFDP: a file transfer application sitting atop a general infrastructure providing routing and reliability.

**Figure 4-4: A CFDP Evolution Path to Use DTN as the CFDP Unitdata Transfer Service**

Many application and middleware protocols use a message-based communications model, where Application layer PDUs are exchanged over the network. Spacecraft commanding, for example, could be implemented using a messaging model. While spacecraft operations may use a file-oriented model where sequences of spacecraft commands are uplinked, checked, and executed as blocks, there will likely be cases where single commands are warranted.

One example of a message-based communications service is the Asynchronous Messaging Service (AMS). AMS requires an underlying transport service, such as the DTN data delivery service.

## 4.3 DESIRABLE CHARACTERISTICS

The following are not necessarily requirements on the network service provided, but are highly desirable characteristics.

**Low Latency**—The network service should impose minimal latency in addition to the physical transmission latency of the path when the path is connected end-to-end.

**Low Overhead**—The network service should not impose more than a reasonable amount of per-packet overhead.

# 5 CANDIDATE SPACE INTERNETWORKING TECHNOLOGIES

## 5.1 OVERVIEW

This section examines in more detail the following candidate technologies for use as a space internetworking layer:

– custom data forwarding;

– CCSDS Space Packets;

– CCSDS File Delivery Protocol (CFDP);

– Internet Protocol (IPv4/IPv6);

– the Bundle (DTN) Protocol.

## 5.2 CUSTOM DATA FORWARDING

### 5.2.1 GENERAL

Custom data forwarding mechanisms can be constructed on an ad-hoc basis. The benefits of such approaches are that they only need be developed when needed, and they can be point-solutions tailored to specific mission requirements and constraints. This should allow for more efficient solutions, although the current Mars relay architecture does not fall into this category because of its reuse of much of the direct-to-Earth data forwarding chain in the relay path.

The primary drawback of custom forwarding is that there is no guaranteed interoperability or opportunity for cross support. Further, the design and maintenance costs of such approaches would need to be borne by each mission, and replicated every time forwarding was needed. The lack of a common, standardized solution means that there is no opportunity for long-term international effort to debug and mature the technology. Finally, construction of end-to-end data paths spanning multiple hops might involve several different custom forwarding solutions and would likely be very complex, expensive, or both.

As an example, the Mars Exploration Rovers and the Mars Phoenix lander currently use an ad-hoc mechanism to forward data between Earth and landed elements on the surface of Mars. The basic mechanism uses the Proximity-1 data link between landed assets and an orbiter (generally Mars Odyssey but Mars Global Surveyor and Mars Express have also been used), and the CCSDS Telecommand/Telemetry protocols between the orbiter and the Earth. This approach, while a great leap forward, is inefficient and suffers from many of the drawbacks above. In particular it does not provide an interoperable internetworking capability, and it would be very complex to use alternate relay nodes or to extend the relay path if that were desirable.

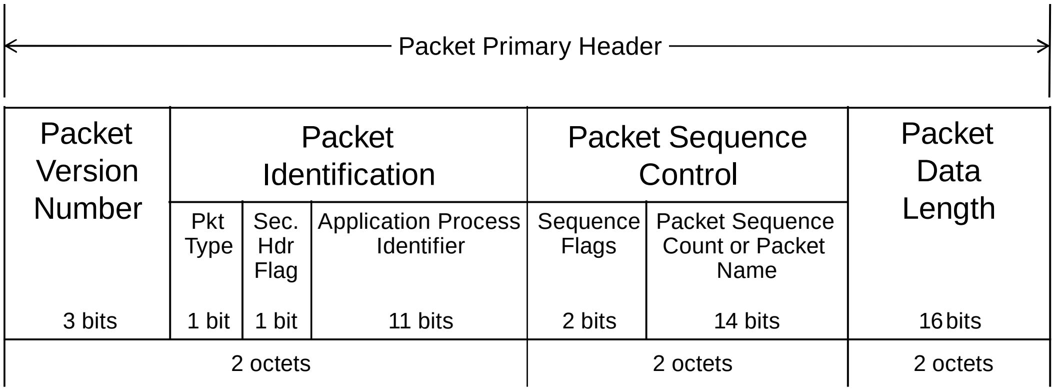
### 5.2.2 CUSTOM DATA FORWARDING: SUMMARY

While custom data forwarding may be very efficient when only one relay hop is involved, its lack of standardization and interoperability mean that attempting to build an automated system, especially one that could (if desired) automatically manage routes among multiple custom systems, would be prohibitively difficult if not impossible.

## 5.3 CCSDS SPACE PACKETS

### 5.3.1 GENERAL

CCSDS Space Packets as specified in reference [9] are an internationally standardized data structure designed to encapsulate Application layer data. Space packets contain an 11-bit application process identifier (APID) to identify the application (in current operation, typically the application that generated the packet), as well as sequencing information, together with the length of the data field. Figure 5-1 shows the format of the Space Packet Primary Header.



Packet Primary Header

| Packet Version Number | Packet Identification | | | Packet Sequence Control | | Packet Data Length |
|---|---|---|---|---|---|---|
| | Pkt Type | Sec. Hdr Flag | Application Process Identifier | Sequence Flags | Packet Sequence Count or Packet Name | |
| 3 bits | 1 bit | 1 bit | 11 bits | 2 bits | 14 bits | 16 bits |
| 2 octets | | | | 2 octets | | 2 octets |

**Figure 5-1: CCSDS Space Packet Primary Header Format**

CCSDS Space Packets can be used to form the basis of a multi-hop data forwarding architecture as shown in figure 5-2.

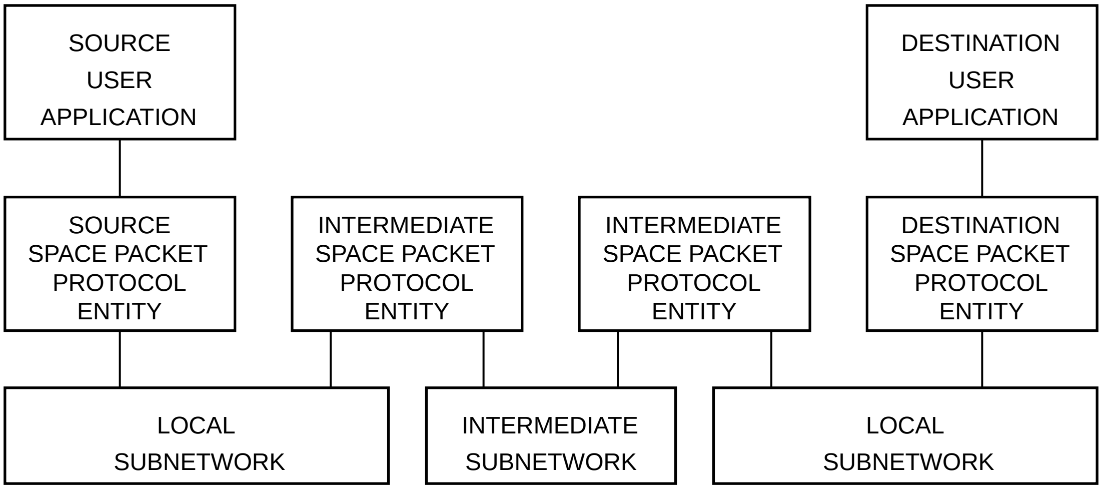

SOURCE USER APPLICATION
DESTINATION USER APPLICATION
SOURCE SPACE PACKET PROTOCOL ENTITY
INTERMEDIATE SPACE PACKET PROTOCOL ENTITY
INTERMEDIATE SPACE PACKET PROTOCOL ENTITY
DESTINATION SPACE PACKET PROTOCOL ENTITY
LOCAL SUBNETWORK
INTERMEDIATE SUBNETWORK
LOCAL SUBNETWORK

**Figure 5-2: Managed Multi-Hop Network Based on Space Packets**[1]

There is currently no standard for the functions, control, or behavior of the intermediate Space Packet Protocol entities in figure 5-2.

### 5.3.2 FEATURES OF SPACE PACKETS

In the Space Packet Protocol, Logical Data Paths (LDPs) define the paths taken by packets as they traverse subnetworks. LDPs are defined by the (APID) in the Space Packet header together with an optional APID qualifier such as the combination of the spacecraft identifier (SCID) and transfer frame version number. An LDP defines a unidirectional path from source to destination through the set of intermediate links.

A number of issues would arise if one attempted to use Space Packets as an internetworking layer:

- The APID qualifier, needed to disambiguate APID namespaces, is not part of the Space Packet Protocol data structure; it is usually carried by a protocol or protocols of the underlying subnetworks. This is problematic if not all of the subnetworks in the path support compatible APID qualifiers. For example, if one or more space agencies were to deploy non-CCSDS links such as WiMax on the Moon or on the surfaces of other planets, special mappings would have to be constructed to carry the APID qualifier information across the non-CCSDS links.
- Because Space Packets use a single SCID/APID pair, this pair needs to function as a path identifier that identifies both the source and the destination of the data. A

[1] Figure 2-2 from the CCSDS 133.0-B-1, Space Packet Protocol.

second SCID/APID pair would be needed to identify the reverse path. Thus it is not possible to identify both the sender and the receiver of information when using Space Packets.

– If the APID qualifier is the master channel identifier (as is *required* when Space Packets are used over CCSDS TC/TM and AOS links), then there is a further issue with addressing: does each frame for intermediate hops carry the SCID of the destination spacecraft or the next hop?

  - If it carries the SCID of the destination, then possibly multiple spacecraft might receive copies of the frame. This might be the case if there were multiple orbiters around Mars and the frame were destined for a particular lander, for example.

  - If it carries the SCID of the intermediate destination (next hop), then APID space would need to be allocated for every destination application reachable through that next hop. This would be difficult given the limited APID space (11 bits).

  - The Space Packet Protocol service interface allows the specification of a QoS requirement to be used to select the appropriate QoS in the subnetworks along the LDP. While this might work for the initial subnetwork, the QoS indication is not signaled in the Space Packet Protocol itself and so it is unclear how QoS requirements can be applied to subsequent subnetworks.

Because the Intermediate Space Packet Protocol entities in figure 5-2 are defined only in that they forward Space Packets based on LDP information that is completely managed, they provide little benefit in terms of reducing operations complexity or providing interoperability. In particular, one agency’s mechanisms for managing the mappings between addressing information (APID/APID Qualifier) and path could be completely different from another agency’s. This would make managing multi-hop paths involving multiple agencies difficult.

## 5.4 CCSDS FILE DELIVERY PROTOCOL (CFDP)

### 5.4.1 GENERAL

The emergence of file based operations resulted in an effort by CCSDS to define an internationally standardized reliable data transfer protocol which was developed into the CCSDS File Delivery Protocol (CFDP) (reference [7]). CFDP provides for the optionally reliable delivery of files across multiple hops in space. The prime benefits of CFDP are that it is internationally standardized (as a CCSDS Recommended Standard), it has a proven flight heritage, and it provides a store-and-forward relaying capability. CFDP is designed to run over a Unitdata Transfer layer (UT layer) that mediates between the file transfer mechanisms of CFDP and an underlying data transport mechanism. For example, CFDP can use different UT layers to transfer CFDP blocks over CCSDS Telemetry/Telecommand (TC/TM) links or Internet protocols such as the User Datagram Protocol (UDP).

### 5.4.2 CFDP FEATURES

The main features of CFDP are:

- file transfer mechanism, including metadata associated with files;
- reliable/unreliable file transfer modes;
- multi-hop file transfer using CFDP extended procedures and/or store-and-forward overlay (SFO) service (not globally implemented).

Figure 5-3 shows an example of CFDP operation taken from the SISG report (reference [11]). Here applications build files that are sent via command to various destinations, in this case the relay orbiter or a landed element on the surface of Mars. In this example, a CCSDS TC is used to transfer files across the long-haul space link between the Earth and the orbiter, and the Proximity-1 Data Link layer is used between the orbiter and the landed asset.

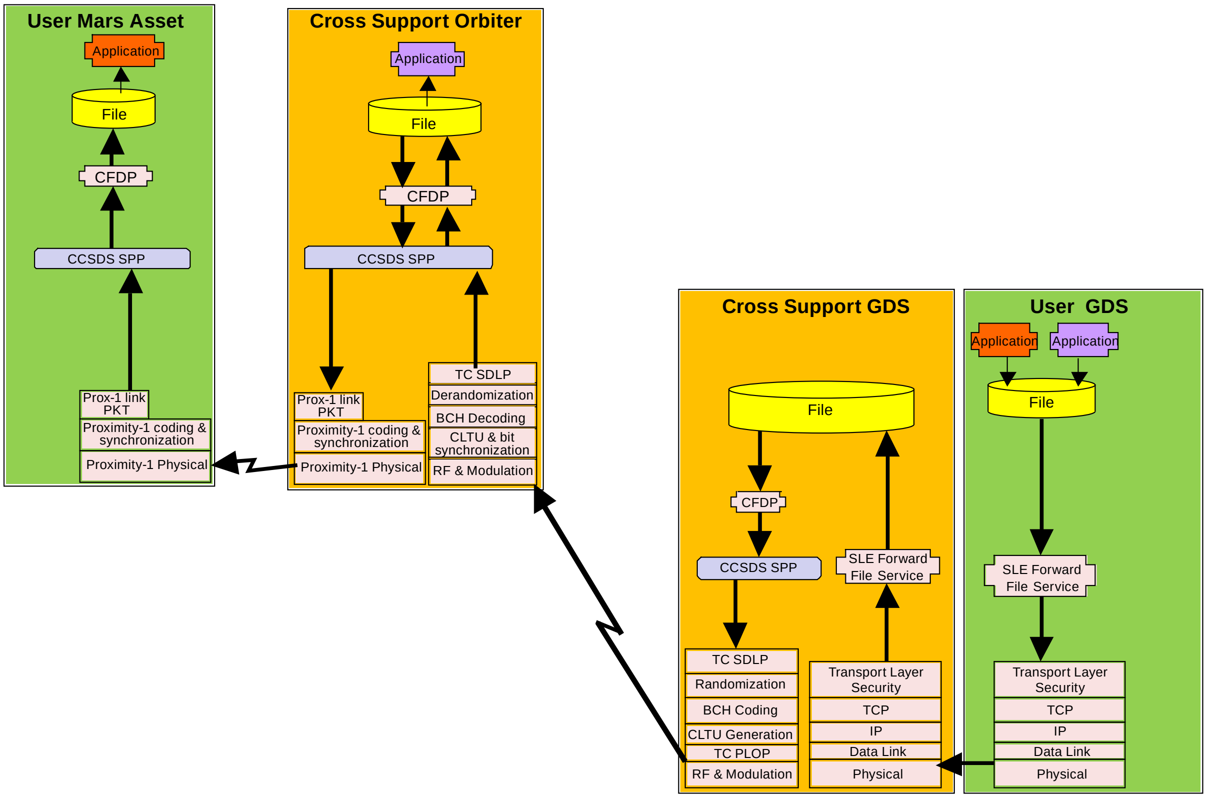


**Figure 5-3: Forward Link Example of Mars Scenarios—End-To-End File Delivery**

The approach taken to CFDP definition was to analyze the existing manual procedures which were used within the various agencies' operations segments and to assimilate these into an automated process. The resulting protocol consisted of two components:

- A reliable data transfer engine incorporating negative acknowledgement, retransmission, conglomeration (the ability to request retransmission of multiple lost data segments in a single request), progress reporting, and suspend/resume/cancel mechanisms to allow operation in noisy, long delay, and disrupted transmission environments.

- A set of file manipulation primitives which can be used to manipulate remote filestores by operations such as file and directory creation, deletion, naming, and, most importantly, copying.

In addition, to support relaying of file transfers through intermediate waypoints such as planetary orbiters, CFDP incorporates procedures for custody transfer of data (the so-called 'extended procedures'). The limitations of these procedures lie in the lack of a guarantee of end-to-end delivery.

A further mechanism, store-and-forward overlay (SFO), takes advantage of the proxy file transfer capability of CFDP and provides end-to-end accounting. This is provided at the expense of requiring whole files to be taken custody of at each waypoint and not allowing opportunistic streaming of individual segment of file data by the waypoints.

Architecturally, both SFO and the extended procedures are similar, both being users of the 'core' CFDP procedures.

To scope CFDP, a number of strawman scenarios were developed. CFDP is capable of satisfying these scenarios except for those involving taking advantage of multiple parallel data paths for a single file transfer operation. This could occur, for example, if a lander were to be serviced by two orbiter relays which could cooperatively deliver data. It could also occur in ground station networks where one station could autonomously hand over data transmission to another when the first station's communication contact with the spacecraft is broken.

CFDP provides four methods of data retransmission which are:

- Deferred, where retransmissions occur after the whole of the file has been attempted to be sent. After extensive prototyping experience this appears to be the preferred mode.

- Immediate, where retransmissions occur immediately after discontinuities in segment sequence numbers are detected at the receiver.

- Prompted, where the transmitting end asks the receiver to report on missing segments.

- Asynchronous, where an external stimulus (e.g., a timer or a notification of imminent link shutdown) triggers retransmission requests at the receiver.

CFDP provides a standalone capability for file manipulation, reliable copying, and limited store-and-forward (multi-hop) file copying.

## 5.5 THE INTERNET PROTOCOL (IPV4/IPV6)

The Internet Protocols (IPv4/IPv6) provide Network layer addressing of data independent of the data links used. The IP protocol suite (including IP and the associated protocols such as routing, domain name service, etc.) is very mature and well-understood terrestrially, but makes some assumptions that limit its utility as a fully general space internetworking protocol. Specifically, the bulk of experience with the IP suite assumes well-connected end-to-end paths, while mature terrestrial IP routing protocols assume relatively stable network topologies. Some other aspects of the IP suite, such as resolving Domain Name Service (DNS) names to IP addresses, assume low latencies as well as connectivity. Where these assumptions can be made to hold or where static tables can be used in place of network lookups, such as in near-Earth (and possibly out to lunar) environments, the IP suite, including commonly-used IP-based applications, can be used.

The Internet suite of protocols is ubiquitous in terrestrial networking.

The main features of the Internet Protocol are:

– the IP suite provides unreliable delivery of datagrams with possible differentiated service;

– IPv4 supports in-network fragmentation and reassembly of fragmented datagrams at the destination; IPv6 requires fragmentation to take place at the source;

– the IP suite comprises a mature set of protocols for security, network management, and routing;

– IPv4 implementations require a contemporaneous end-to-end path from source to destination in order to deliver data;

– transport protocols (e.g., TCP, SCTP) can be used to provide reliability on top of IP services.

The largest issue with deploying IP is the assumption of an end-to-end path. If the network is partitioned so that there is no current path from one part of the network to another, IP datagrams that reach the partition boundary will have nowhere to go and will be discarded. The standard transport protocol used for reliability with IP, the Transport Control Protocol (TCP), also suffers moderate to severe performance degradation as round trip times and error/loss rates increase.

It might seem attractive to write a custom implementation of the IP protocols that stored packets when no end-to-end path was available as a way to leverage the large body of work in the IP suite. Unfortunately, other protocols besides just the datagram forwarding depend on end-to-end connectivity and low delays. For example, the more mature routing protocols for IP networks exchange information in order to build a graph of the current connectivity and then to route datagrams on that graph. In disconnected environments, these protocols will not function well. Reactive IP routing protocols for mobile ad-hoc networks typically need to probe to establish new paths, which could involve long delays before data could be transmitted. Thus the large body of supporting protocols for IP cannot be directly leveraged for space internetworking in environments that may contain disruptions and temporary network partitions.

## 5.6 THE BUNDLE PROTOCOL

### 5.6.1 GENERAL

RFC5050 defines a DTN protocol known also as the Bundle Protocol after the name given to RFC5050 network PDUs. RFC5050 defines the rules for formatting Bundles for transmission between DTN nodes, and the requirements for processing and responding to administrative flags and messages. Figure 5-4 shows the format of RFC5050-compliant Bundle headers.

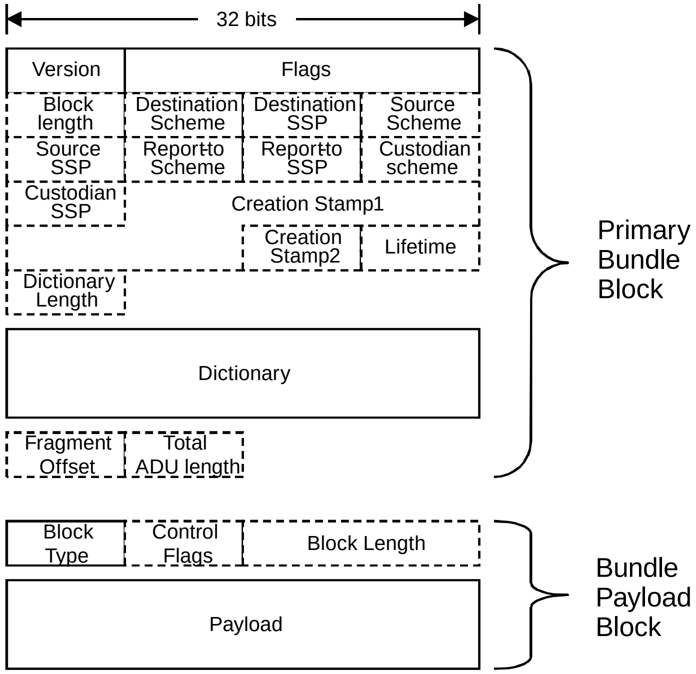


**Figure 5-4: Bundle Protocol Blocks**

As with the Internet Protocol, there are supporting protocols needed to operate an internetwork based on BP. For example, to improve performance when reliable data delivery is required, BP strongly desires a reliable hop-by-hop service from the underlying layer. The Licklider Transmission Protocol has been developed to provide this service. A network management mechanism and associated protocols are being developed to support configuration, operations, monitoring, and provisioning of internetworks based on BP. In the future, one or more dynamic routing protocols could also be developed.

Figure 5-5 shows how BP would operate in an end-to-end data transfer between a mission control center and a Mars surface asset. In the terrestrial Internet between the mission control center and the ground station, BP can be deployed as an overlay network on top of

TCP. Practically this means that BP may be present only at the mission control center and the ground station, relying on IP to connect the two. At the ground station, BP may store messages until the next contact period with the relay satellite. A custody transfer acknowledgement from the ground station would inform the control center that the messages had been successfully received and queued for transmission.

When transmitting messages across the space link, BP would probably use a different set of Data Link and Transport layer protocols than were used for the Internet connection. The figure shows BP using LTP, CCSDS Encapsulation Packets, and the CCSDS AOS data link. Again, messages marked for reliable delivery may be stored on the orbiter and acknowledgements sent to the ground station at the next opportunity. This way, if messages are lost in transit between the orbiter and the rover, they can be retransmitted from the orbiter instead of having to go back across the deep-space link.

Finally, the orbiter can use the Proximity-1 data link protocol to send messages to the rover.

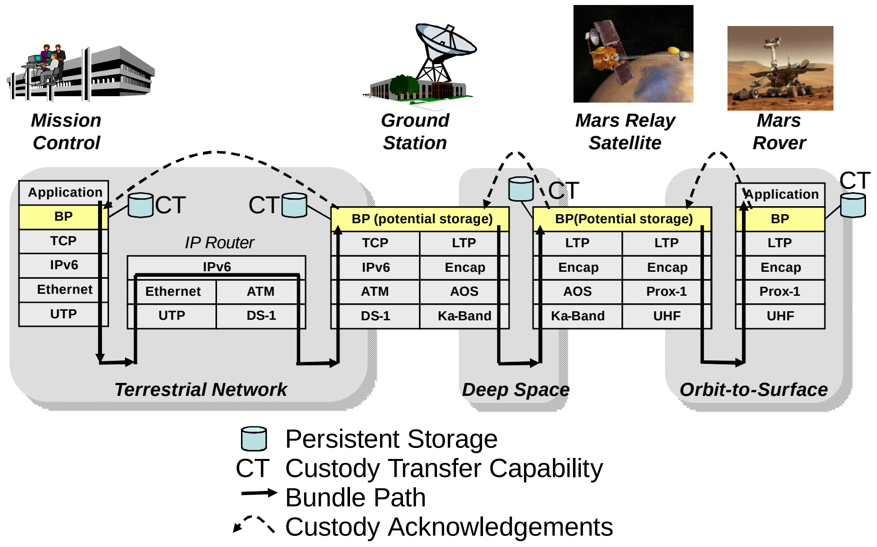


**Figure 5-5: DTN Used for End-to-End Data Transfer to a Mars Rover**

If, during a communications pass, some new command messages that were not transmitted to and stored at the ground station need to be sent, mission control can transmit the messages during the contact. Depending on the priorities of the various messages, the new messages from mission control might be transmitted before messages queued at the ground station, or they might be placed into a FIFO queue for transmission once all of the queued messages have been sent.

### 5.6.2 SERVICES PROVIDED BY THE BUNDLE PROTOCOL

The Bundle Protocol provides the following services:

- Completeness: The Bundle Protocol provides an atomic, message-oriented delivery service with no notion of sequenced delivery. Individual messages are delivered (or not) in their entirety. A Bundle is not delivered with gaps.

- Error-free data delivery: The Bundle Protocol as currently defined may deliver data with errors *if* the end-to-end data integrity mechanisms specified in the Bundle security protocol are not invoked. If end-to-end integrity is not used, the Bundle Protocol relies on the hop-by-hop reliability mechanisms of the individual underlying transport mechanisms. If end-to-end integrity is used, Bundles are guaranteed to be delivered error-free.

- Delay/disruption tolerant data delivery. If DTN PDUs reach a point in the network path where no forward progress can be made (because, e.g., the next-hop data link is not available), DTN may store the PDUs while waiting for the next-hop link to become available.

- Flexible naming/addressing scheme. DTN uses Uniform Resource Identifiers (URIs) (reference [22]) to identify the endpoints of communication. In addition to traditional '(host, port)'-type addressing, these URIs allow data to be addressed to users that meet some criteria, such as all sensors that have registered an event within the past hour.

- Time-to-live. Each Bundle is assigned (by the source application) a 'time-to-live' that is meant to reflect the useful lifetime of the data. The time-to-live represents an actual time duration, not a network hop count, and is used to remove Bundles from the system if they cannot be delivered in a timely manner.

- Optionally reliable data transfer. DTN implements reliable data delivery by means of in-network checkpointing of Bundle progress called custody transfer.

- Per-Bundle Control Flags. Each Bundle contains a set of flags that can trigger particular status reports about the Bundle's progress. These include:

  - A Bundle priority field that allows three levels of priority. RFC5050 does not specify how these levels or priority are handled. CCSDS may want to specify strict priority queuing with or without preemption, for example.

  - Optional end-to-end confirmation of Bundle delivery. Applications may request that a confirmation of delivery of the Application layer data be sent to a particular DTN Endpoint Identifier (EID) (the 'report-to' EID, see below). This provides an indication to the report-to entity that the destination application received the data and is generated when the data is passed to the destination application. In particular, this notification is not an indication that the destination application actually processed the received data.

- Request reporting of Bundle reception. If reporting of Bundle reception is requested by an application, intermediate nodes in the path will generate notifications to the Bundle's 'report-to' EID when the Bundle is received at each node.
- Request reporting of custody acceptance.
- Request reporting of Bundle forwarding.
- Request reporting of Bundle delivery.
- Request reporting of Bundle deletion.

NOTE – The reports can be used to provide data accountability for Bundles.

– Alternate 'Report-To' Addressing. The reports generated by a Bundle may be directed to a different destination than the source. Reports may be directed towards destinations that are not generally reachable so that data accountability reports could be generated at nodes but would not be transmitted unless specific action were taken to retrieve the records.

### 5.6.3 ADDITIONAL FEATURES OF THE BUNDLE PROTOCOL

In addition to the services provided above, the Bundle Protocol also supports:

– Fragmentation. Bundles may be split inside the network and reassembled later before being delivered to the destination(s).

– Extensibility. Bundle Protocol data units are composed of a variable number of 'blocks'. Block types are identified by Self-Delimiting Numerical Values (SDNVs) so that expression is both efficient and highly extensible. Each block carries with it a set of flags identifying how nodes that do not understand the block should treat it (pass it unmodified, remove the block, discard the Bundle, etc.). Thus additional capabilities such as 'keep at most *N* of this type of cyclic telemetry value' can be implemented.

### 5.6.4 SERVICES NOT SUPPORTED BY THE BUNDLE PROTOCOL

The following services are NOT supported by the Bundle Protocol as specified in RFC5050:

– In-order delivery. BP's model for application interaction is via atomic, Application layer messages. BP does not attempt to maintain the order of messages submitted to the network, and indeed this may run counter to the priority markings of a sequence of messages. That is, if several messages are sent with low priority followed by a message with high priority, it is likely that the high priority message will arrive before at least some of the lower-priority ones.

- Duplicate suppression. BP does not specify that duplicate messages be suppressed before delivering them to the destination. The information that would be needed to implement duplicate suppression is already present in the primary Bundle block; the BundleID, the combination of the source EID, the sending timestamp, and the sending sequence number, is guaranteed by the protocol to be unique for each Bundle. Thus duplicate suppression could be implemented by removing Bundles with duplicate BundleIDs. This would require maintaining state, at least at receivers, to keep track of the BundleIDs that had already been delivered. Such state would have to be maintained at least for the lifetime of the Bundle, and possibly longer.

- Combined elements of ISO layers 3 (Network) and 4 (Transport). The Bundle Protocol does not provide any Application layer services such as file transfer, messaging, voice, or video transfers. Such services can be constructed using the services of the Bundle Protocol, similarly to how they are constructed using the Internet Protocol in connected environments. Annex B1 describes how the file management capabilities of CFDP can be implemented over the networking capabilities of the Bundle Protocol, thus allowing CFDP to take advantage of multiple parallel paths during a single file transfer.

### 5.6.5 LICKLIDER TRANSMISSION PROTOCOL

While the Bundle Protocol makes very few demands of the underlying communication system, there are features that, if implemented beneath DTN, can improve performance and efficiency. These include:

- Reliability. By providing reliability at the data link level, LTP can more accurately gauge expected round trip times and so make better decisions about when to proactively retransmit data that may have been corrupted in transit.

- Data Unit Size Management. One of the problems with early deployments of CFDP was acknowledgement channel bandwidth. If many small files were transmitted using CFDP, the acknowledgements needed could consume the entire return channel. LTP can aggregate many small SDUs into a larger LTP segment for transmission, thus reducing the volume of acknowledgement traffic (since fewer, larger LTP segments are acknowledged). Segmenting a DTN PDU into multiple smaller segments for transmission across the wire can also provide more efficient retransmission in the case of data loss.

## 5.7 CONCLUSIONS

It is believed that the Bundle Protocol as specified in RFC5050 is best-suited to support in-space relaying/internetworking. In particular, bundling supports the requirements of section 4, including PDU delivery in the presence of possible network partitions and/or simplex links/paths, ability to address PDUs to particular applications, data accountability, reliability, and security.

It is acknowledged that the overall architecture for space internetworking will probably involve build-out of Space Packet-based services as well as IP-based services in addition to Bundle-based services.

At this time Internet Requests for Comments (RFCs) exist for the basic Bundle Protocol and the above-described optionally reliable hop-by-hop transmission protocols that are intended to be adapted for use in space.  Additional support protocols such as security and network management need additional development and standardization within CCSDS.

# 6 CONCLUSION

The Space Internetworking Strategy Working Group (SISG) evaluated the merits of various approaches to transitioning to space internetworking during the 2015-2020 timeframe, including using the candidate architectures described above. The SISG came to the consensus conclusion that the two most viable approaches for transition are:

a) continue with the current baselined mechanisms for 2015-2020 (essentially maintaining the current one-hop packet-over-link model with any data relaying being provided by custom means);

b) transition to and end-to-end internetworked system that uses a single internetworking layer such as the Bundle Protocol.

It is believed that an internetworked approach based on the Bundle Protocol as described above will lead, over the next several decades, to lower operations costs, higher data return, and a more flexible system that can more easily support the envisioned growth in the number of spacecraft.

# ANNEX A

# COEXISTENCE OF DTN, IP, AND SPACE PACKETS

The DTN data delivery service will need to coexist with other protocols, including at least CCSDS Packets, and probably with Internet Protocols in parts of the network.

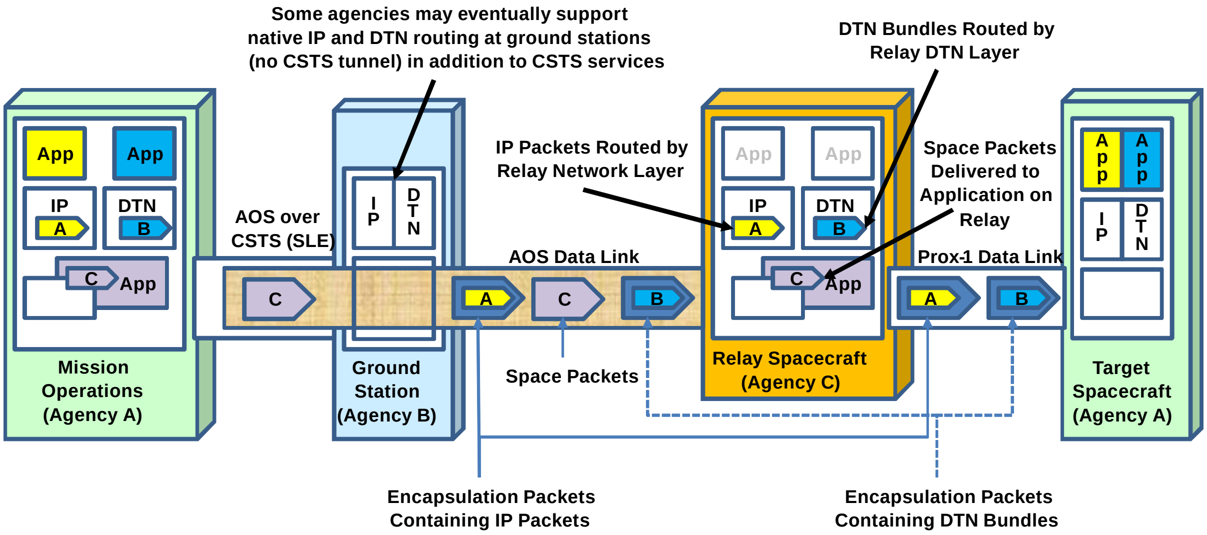

Some agencies may eventually support native IP and DTN routing at ground stations (no CSTS tunnel) in addition to CSTS services
DTN Bundles Routed by Relay DTN Layer
IP Packets Routed by Relay Network Layer
Space Packets Delivered to Application on Relay
App
App
IP
A
DTN
B
C
App
Mission Operations (Agency A)
AOS over CSTS (SLE)
C
I P
D T N
Ground Station (Agency B)
AOS Data Link
A
C
B
Space Packets
Relay Spacecraft (Agency C)
Prox-1 Data Link
A
B
App
App
Target Spacecraft (Agency A)
Encapsulation Packets Containing IP Packets
Encapsulation Packets Containing DTN Bundles

**Figure 6-1: Example Showing Coexistence of Space Packets, IP Packets and DTN PDUs on Space Links**

Figure 6-1 illustrates the potential coexistence of Space Packets, IP packets, and DTN Bundles in a system where Agency A uses the network service provisioned by Agencies B and C to communicate with a target spacecraft. IP packets are shown here not because they are particularly well-suited to deep-space missions, but because they may be appropriate for certain near-Earth (including lunar) missions.

In the figure, Space Packets (C) are not relayed; they can be transmitted only over a single data link connection. They are shown tunneled over existing Cross Support Transfer Services (CSTS) between the mission operations center and the relay spacecraft, via the ground station. Space packets are shown terminating in a Space Packet application on board the relay spacecraft (assuming for the moment that Agency C were willing to accept packets from Agency A). For simplicity, both IP packets and DTN Bundles are also shipped over the CSTS tunnel, though future modifications to the CSTS service could allow IP packet and DTN Bundles to be multiplexed into the data stream at the ground station.

Both IP packets and DTN Bundles are 'Network layer' data structures that can be forwarded across multiple hops. This is illustrated in the figure by the IP and DTN boxes sitting above the Data Link layer. To traverse CCSDS data links, while IP packets could be encapsulated directly in CCSDS data links, both IP and DTN are shown inside CCSDS Encapsulation Packets. Notionally the IP and DTN boxes on the relay spacecraft forward packets or

Bundles between the incoming and outgoing data links. IP-based and DTN-based applications may be resident on the relay spacecraft as well, as illustrated by the yellow and blue ‘Application’ boxes.

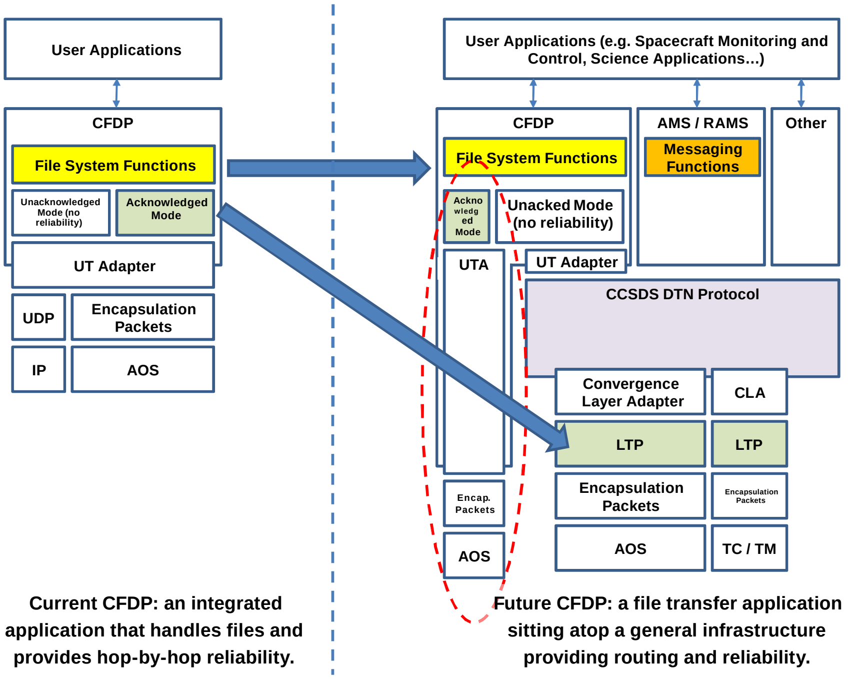
# ANNEX B

# EXAMPLES OF SERVICES THAT MAY BE IMPLEMENTED OVER THE INTERNETWORKING LAYER

## B1 CFDP OVER DTN

### B1.1 GENERAL

Because of CFDP's importance to the space community, it is particularly important to understand how CFDP can function in the internetworked environment. Figure 4-4, reproduced below, describes how the functionality of CFDP can be partitioned into File System Functions, an Internetworking Protocol (e.g., DTN), and Data Link Operations. Refactoring CFDP over a reliable Network layer has a number of advantages, including support for CFDP scenario 4 with different parts of a file routed along different and possibly parallel network paths to the destination.



Current CFDP: an integrated application that handles files and provides hop-by-hop reliability.

Future CFDP: a file transfer application sitting atop a general infrastructure providing routing and reliability.

Figure B-1: A CFDP Evolution Path to Use DTN as the CFDP Unitdata Transfer Service

### B1.2 NOTIFICATION OF FILE DELIVERY

If the reliability mechanisms of the underlying internetwork protocol (e.g., DTN) are used, CFDP's reliability mechanisms are not required and could be omitted to improve efficiency. This would allow Class-1 (unreliable) CFDP to be used, ***except*** that Class-1 CFDP would not inform the sender when the file was delivered. To provide 'Class-2-like' service, one could use the standard Class-1 CFDP service with an additional Application layer 'received' indication to the CFDP sender.

### B1.3 RELIABILITY MECHANISMS AND STORAGE

Using a modified CFDP implementation as discussed above would leave intact reliability mechanisms at both the internetwork and Data Link layers. While each of these layers might need to maintain copies of data until it is acknowledged, this does not necessarily mean that multiple copies of the data have to be maintained. For example, if a user generates a file and then invokes CFDP's transmission mechanism, it may be possible for the implementations of CFDP, BP, and LTP to pass references to the data without actually making multiple copies. One such mechanism is used in the ION protocol suite, where Zero-Copy-Objects (ZCOs) are used to maintain references to data. From the ION Design and Operation guide v1.5:

> ION's ZCO system leverages the SDR system's storage flexibility to enable user application data be encapsulated in any number of layers of protocol without copying the successively augmented protocol data unit from one layer to the next. It also implements a reference counting system that enables protocol data to be processed safely by multiple software elements concurrently; e.g., a Bundle may be both delivered to a local endpoint and, at the same time, queued for forwarding to another node – without requiring that distinct copies of the data be provided to each element.

This is illustrated in figure B-2. Here the user generates a file and invokes CFDP's send operation giving the filename as a parameter. The CFDP implementation generates a set of ZCOs for pieces of the file and sends those using the Bundle Protocol. The Bundle Protocol custody transfer mechanisms are required to maintain 'copies' (or ZCO references in this case) of the data in case it has to be retransmitted. Below the Bundle Protocol, LTP may be invoked to provide reliable communications over unreliable links, in which case it too must keep references to the data. However, using ZCOs means that only a single copy of the data needs to be maintained, not four.

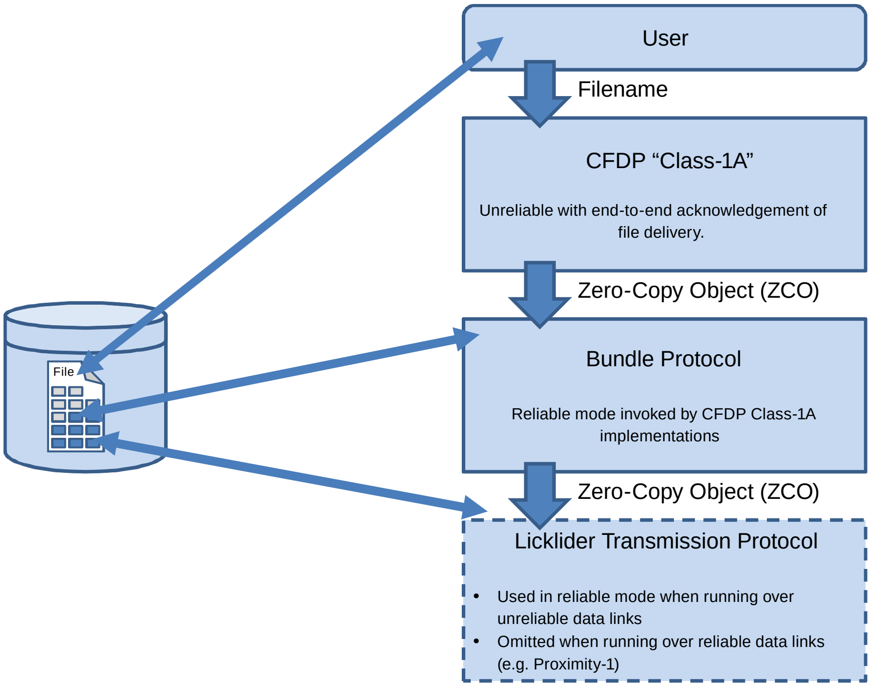

User
Filename
CFDP "Class-1A"
Unreliable with end-to-end acknowledgement of file delivery.
Zero-Copy Object (ZCO)
Bundle Protocol
Reliable mode invoked by CFDP Class-1A implementations
Zero-Copy Object (ZCO)
Licklider Transmission Protocol
• Used in reliable mode when running over unreliable data links
• Omitted when running over reliable data links (e.g. Proximity-1)
File

**Figure B-2: CFDP over BP over LTP Using Zero-Copy-Objects**

In general, the implementations of the various protocol layers each require access to both persistent (e.g., disk) and ephemeral (e.g., memory) storage.

Persistent storage is needed if the implementation needs to ensure that data are maintained across reboots (planned or unplanned) of the system. The persistent storage can take a number of forms depending on the implementation. For example, a single addressable array of octets in a solid-state data recorder or a pre-allocated file accessible via the SOIS File Access service [21] may be allocated for network operations, with the various layers using zero-copy objects to pass pointers to data between layers. In this case the network stack would be responsible for managing the internals of the storage, adding and deleting user content as appropriate.

If a more capable set of file services such as the SOIS File Management services are available, the networking implementation can make use of those to manage persistent data. In this case the networking implementation may create and delete files as well as simply modifying their contents. This would presumably simplify the implementations of BP and LTP. The disadvantage to this approach is that it admits the possibility that the network stack could consume more file resources than it is supposed to, which might be mitigated by a quota capability in the file store implementation.

Sometimes file systems become corrupted to the point that they are unusable. In this case, a networking implementation that relied on a correctly functioning filestore would itself become unusable. While it is desirable to commit bundles to persistent store for reliability, it is not required and some bundles may be manipulated solely in memory (ephemeral storage). For example, bundles that are not forwarded using a reliable mechanism (e.g., bundles forwarded using the unreliable LTP service) that the node does not accept custody of need not be committed to persistent store. This would provide a mechanism for communicating with a node whose file store was corrupt, for example. Alternately, low-layer commands as described in Section 4.2.3 could be devised to attempt to recover the file store. This might provide a more robust solution, since if the filestore is corrupted and the network layer is restarted, even the network layer configuration information would be suspect.

### B1.4 FRAGMENTATION AND EFFICIENCY

Using the CFDP/BP/LTP stack, there are several of places where fragmentation and/or aggregation can take place. CFDP and its UT-layer interface to BP dictates the sizes of the Bundles exchanged between CFDP entities. While BP may fragment Bundles for transmission through the network, it must reassemble them completely at the destination before delivering them to the application. BP cannot aggregate multiple Bundles together into larger data units for transmission.

LTP may use a segment size that is smaller than the Bundles produced by CFDP, or it may aggregate multiple Bundles together from the standpoint of its reliability accounting. This allows LTP to be tuned to the specifics of a particular data link, including any link asymmetry. In particular, this aggregation mechanism allows LTP to provide reliability while limiting the traffic needed by acknowledgements.

## B2 MECHANISMS TO SUPPORT EXISTING PACKET-BASED APPLICATIONS OVER THE DTN SERVICE

It may be desirable to be able to support current missions using the DTN network infrastructure described above. To do so would require the specification and standardization of bitstream- and/or CCSDS packet-based *proxy applications* that can convert between existing (non-DTN) and Bundle Protocols. Essentially, such proxy applications would serve as *tunnels* for the existing protocols over DTN. This would allow the extension of Space Packets and or arbitrary bitstreams across multiple hops. The main disadvantage of such proxies would be that they would require a large amount of configuration to convert between addressing formats and parameters of the various protocols.

Many existing space applications use CCSDS Packets to organize and transfer data, and may want to continue to use Space Packets as Application layer data units even as they want to move to a multi-hop, internetworked environment. It may also be desirable to have a similar service capable of delivering CCSDS Encapsulation Packets across multiple hops to an end system. If these services are indeed desired by the space community, standard DTN services can be developed to support carriage of CCSDS Space and Encapsulation Packets across DTN Networks. This would essentially provide a ‘DTN tunnel’ for CCSDS Space and Encapsulation Packets, allowing the routed DTN infrastructure to provide the common Network layer service for delivering higher-layer data units (in this case CCSDS Space or Encapsulation Packets) to their destinations.

Figure B-3 shows a Space Packet proxy application resident in the host agency's control center. This application receives packets transmitted over a to-be-defined CSTS Packet service and uses DTN to route the packets to the penultimate spacecraft (relay). There the peer packet proxy extracts the packets and presents them to the packet SAP of the last-hop Data Link layer. The letters underneath the facilities indicate that, because of standardization of CSTS and DTN services, the facilities could be operated by different agencies.

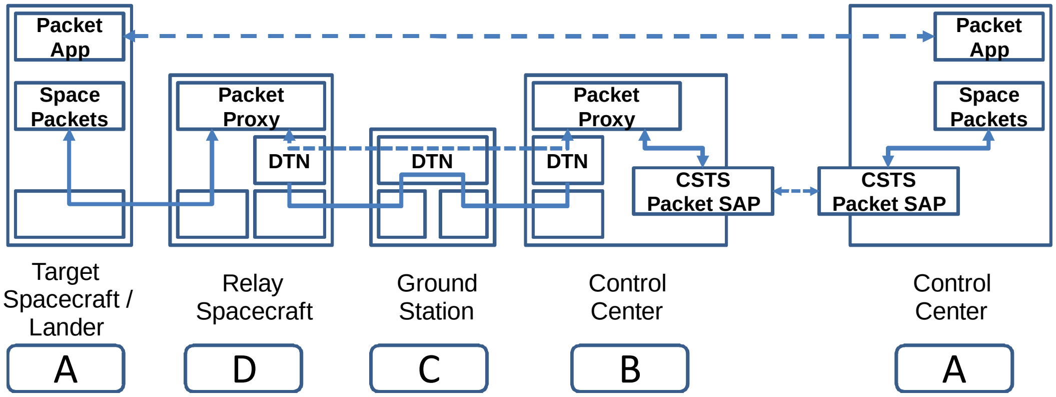


**Figure B-3: Packet Transfer via DTN Tunnel Example**

Alternatively, the packet proxy could be implemented in the guest control center, proximate to the end system application as shown in figure B-4.

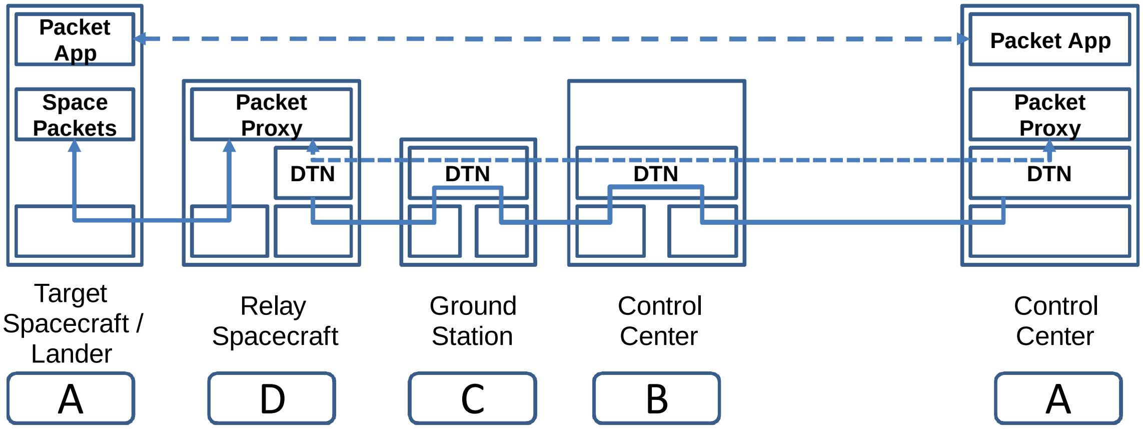


**Figure B-4: Packet Transfer via DTN Tunnel Example**

Packet proxies could be used to support Space Packets or Encapsulation Packets. Alternatively, similar proxies could be defined to support delivery of arbitrary bitstreams (e.g., frames) to destinations. These would allow standardized hardware commanding of remote assets across multi-hop network paths. Many spacecraft implement low-level 'hardware commanding' directly on top of bitstreams or Space Packets. A set of standardized services to move packets or bitstreams across multiple network hops to their destinations would allow these types of low-level commands to be executed over multiple hops and cross-supported among agencies.

Not shown in the above figures is the management interface for controlling the packet tunnels. This interface will need to set up the tunnels, identifying the tunnel endpoint (e.g., the particular application and relay spacecraft at the end of the tunnel). Also, any information needed to invoke the last data link hop such as the protocol options to use, the time to instantiate it, etc., will need to be specified either as part of the packet tunnel interface or as a separate management exchange.

A similar proxy mechanism could transport arbitrary bitstreams across multiple space links. Because arbitrary bitstream transport tends to lead to non-interoperability, such mechanisms are discouraged here.

# ANNEX C

# ADDITIONAL INFORMATION ABOUT THE BUNDLE PROTOCOL

## C1 SERVICES THE BUNDLE PROTOCOL REQUIRES

### C1.1 OVERVIEW

This annex describes the lower-layer and ancillary services required to operate the Bundle Protocol.

### C1.2 COMMUNICATIONS

The Bundle Protocol was designed to support a wide range of underlying network and data link technologies via the ‘Convergence Layer’ (CL) mechanism. Subection 7.2 of RFC5050 lists the minimum requirements of a CL as:

> Each CL protocol adapter is expected to provide the following services to the Bundle Protocol agent:
>
> - sending a Bundle to all Bundle nodes in the minimum reception group of the endpoint identified by a specified endpoint ID that are reachable via the convergence layer protocol; and
> - delivering to the Bundle Protocol agent a Bundle that was sent by a remote Bundle node via the convergence layer protocol.

This essentially means that CLs must be able to send and receive Bundles to and from other DTN nodes that implement compatible CLs. In particular, while it may be desirable to implement such features as fragmentation, aggregation, and reliability in a CL, they are not required. An example of a space CL might be, for example, an implementation of the Licklider Transmission Protocol running over CCSDS Telemetry and Telecommand. Alternatively, DTN might run directly over the Proximity-1 reliable packet delivery service.

### C1.3 TIME

To support Bundle lifetimes as ‘wall-clock’ times-to-live, the Bundle Protocol requires *loose* time synchronization among nodes. Thus the Bundle Protocol requires access to a time source on board the spacecraft that the protocol can then convert into its internal time representation of seconds since midnight, January 1, 2000.

The time-to-live field in the Bundle header is used to remove Bundles that remain in the communications system past their useful lifetimes, and applications are expected to set the lifetime long enough to allow delivery of Bundles to their destinations. Because this delivery latency is not necessarily known ahead of time, and possibly not known at all by the application, it is expected that applications will be liberal in setting their data timeout values. Thus setting a Bundle's timeout value at, say, a minute past the expected useful lifetime of the data is not unreasonable. This would allow for a clock skew of up to a minute among nodes in the DTN network delivering the Bundle.

There is ongoing work examining the possible benefits of redefining the Bundle lifetime field as a 'countdown timer' instead of a delta from the Bundle creation time. If such investigations prove useful, future extensions to the RFC5050 could adopt the new convention, removing the requirement for even loose time synchronization. CCSDS should coordinate with this work to determine its applicability to the space domain.

### C1.4 STORAGE

In addition to communications and timing information, the Bundle Protocol needs to be able to store and retrieve data. Persistent storage is strongly desired if a node wishes to take custody of Bundles. Efficient use of storage may dictate use of references to a single copy of data by multiple protocols. The minimum requirement for operation is random read/write access to a possibly bounded array of octets.

The exact nature of storage is a matter for the particular BP implementation. The DTN2 reference implementation, for example, can be configured to place Bundle contents in files for persistent storage and to maintain Bundle metadata (including the information from the primary Bundle blocks) in a file-backed database. Alternatively, DTN2 can be configured to keep all Bundle payloads entirely in memory. The ION implementation uses a set of libraries that can interface to (and on systems that do not have one, mimic) solid-state data recorders. These mechanisms can be configured to be either file-backed or completely memory based.

An implementation might even use a combination of persistent (SDR-based or file-based) and in-memory storage, committing Bundles that warrant persistent storage (e.g., Bundles the node has taken custody of) to a reliable persistent store and keeping the rest in memory. This might facilitate emergency-mode operations if the reliable data store (if any) backing the BP/LTP implementation was corrupted or otherwise unavailable.

It would be expected that BP and LTP implementations that used file-based mechanisms for storage would use standard CCSDS file interfaces (e.g., SOIS) for those services.

## C2 NAMING OF BUNDLE PROTOCOL ENDPOINTS

The Bundle Protocol allows for rich naming of endpoints via URI syntax. This provides a great deal of power to support concepts such as intentional naming (identifying the characteristics of the endpoint rather than explicitly identifying the endpoint by address) that may not be needed in space communications.

A less powerful but much more compact naming scheme has been proposed that identifies Bundle Protocol applications by the combination of a *node number* and a *service number*. These are akin to an IP address and port number in the IP protocols. This level of addressability will probably suit space applications for a long time to come, and has the added benefit of being highly compressible within the Bundle Protocol via Compressed Bundle Header Encoding (CBHE—see reference [16]). CBHE allows the node and service number of the various Bundle Protocol endpoint identifiers to be directly encoded in the integer offsets within the primary Bundle block of figure 5-4. This removes the overhead of a text representation of the URI and allows the dictionary to be completely removed from the primary Bundle block. CBHE can reduce the size of the primary Bundle block to as little as 27 bytes.

## C3 BUNDLE PROTOCOL FORWARDING AND ROUTING

In the simplest case, Bundle Protocol routers can use static tables to choose next-hop addresses for Bundles based on the Bundles' destination endpoint identifiers. These contents of these routing tables may be completely managed by the mission operations personnel on Earth. This amounts to a set of static, managed forwarding tables.

A slightly more complex routing algorithm would allow Bundle Protocol routers to make decisions based on the destination endpoint identifier and the Bundle's time-to-live field. This approach was explored in reference [10] which takes as inputs a schedule of link connectivity and a set of Bundles and attempts to schedule Bundle transmissions to maximize the number of Bundles delivered before they expire.

Depending on need, more complex dynamic routing protocols akin to dynamic routing on Earth may be deployed. These will probably continue to differ from their Internet analogues in that the Bundle versions will need to deal with scheduled connectivity.

## C4 BUNDLE PROTOCOL NETWORK MANAGEMENT

While mature network management protocols exist for IP, protocols and procedures to manage BP networks are still under development. Thus early missions will have to mange the network manually, much as links are manually managed now. As BP network management develops, it can be deployed into the network and the manual efforts can be scaled back.

## C5 BUNDLE PROTOCOL SECURITY MECHANISMS

The Bundle Security Protocol, currently in Internet draft form, specifies a number of security mechanisms that can be applied to Bundles, including:

> **Bundle Authentication Blocks (BABs)**: Bundle Authentication provides 'hop-by-hop' security, allowing a receiving DTN node to authenticate that Bundles received were indeed transmitted by a known, trusted source. This can be used to prevent, for

example, an intruder who has access to the network from sourcing Bundles that are then forwarded across scarce resources such as deep-space links. BAB security applies only to a single DTN 'hop'; BABs are removed on reception and must be re-generated by the relay before a Bundle is forwarded.

**Payload Integrity Block (PIB):** Payload integrity provides a means for a receiver to determine whether or not a Bundle payload has been modified since it was signed. Signing and checking operations can be carried out at the ends or in the middle of the network, and such operations may be nested.

**Payload Confidentiality Block (PCB):** Payload confidentiality encrypts the Bundle payload so that if an intermediate node that does not have the correct key intercepts the Bundle, the node will not be able to discern the Bundle contents. The node WILL be able to determine any information that is in the clear, such as the source, destination, and report-to addresses for the Bundle.

**Extension Security Block (ESB):** Extension security provides a way of protecting other arbitrary extension blocks.

End-to-end Bundle authentication can be achieved with the PCB and appropriate security policy. Essentially, if a Bundle decrypts correctly using a particular key, then the receiver should be able to unambiguously identify the source of the Bundle.

## C6 DTN SUPPORT FOR REMOTE IN-SITU NETWORKS

### C6.1 GENERAL

It may be desirable to use other networking technologies for 'in-situ' communications. For example, a group of Mars rovers in close proximity to one another might use Internet protocols for local communication among themselves. A set of lunar rovers or a lunar outpost might do the same. The rationale for using Internet protocols locally could be that they are readily available as part of the operating system of the end nodes (e.g., VxWorks), or that the nodes want to take advantage of existing Internet-based applications. Figure C-1 shows such a local 'island' of in-situ networked communications separated from the Earth.

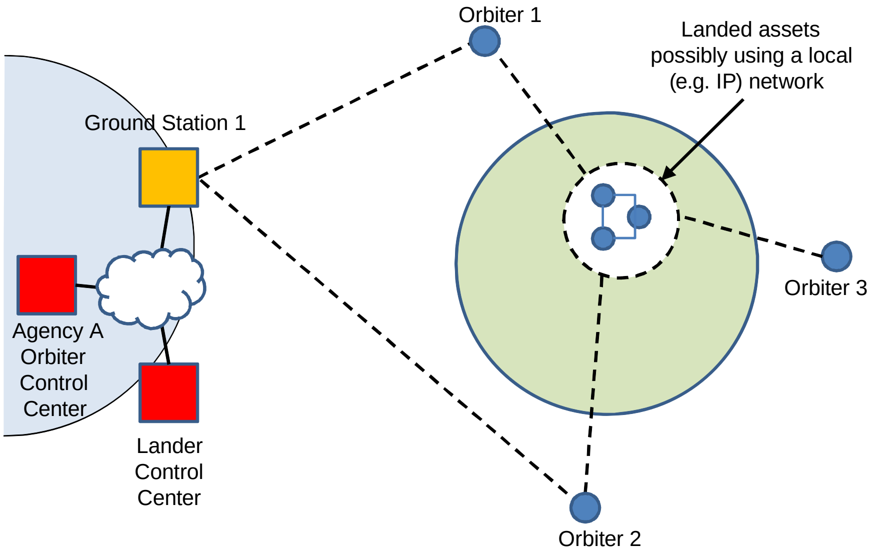

Orbiter 1
Landed assets possibly using a local (e.g. IP) network
Ground Station 1
Orbiter 3
Agency A Orbiter Control Center
Lander Control Center
Orbiter 2

**Figure C-1: In-Situ Network Topology**

If a DTN network protocol is required for communications back to Earth, how do the local communications and the DTN communications interact? How can the local nodes using a non-DTN in-situ networking protocol communicate across a portion of the network using DTN? There are three basic methods through which this can be accomplished:

a) by using native DTN applications and protocols for the ‘in-situ’ and ‘long-haul’ portions of the network;

b) by using Application layer gateways to convert between in-situ networking protocols and DTN;

c) and by tunneling the in-situ protocols inside DTN.

## C6.2 NATIVE DTN APPLICATIONS

In the first method, the applications running in situ simply use DTN even for local communications. The DTN communications might themselves be over Internet protocols such as TCP/IP, but the interface to applications in this model is that of DTN. Here there is no issue interfacing between in-situ and long-haul communications, as all communications are using the Bundle Protocol. Because the Bundle Protocol can switch between using IP-based or CCSDS-based (or other) mechanisms for its hop-by-hop communications, it can easily move from a local IP cloud to communicating over CCSDS AOS or TC/TM for the long-haul communications.

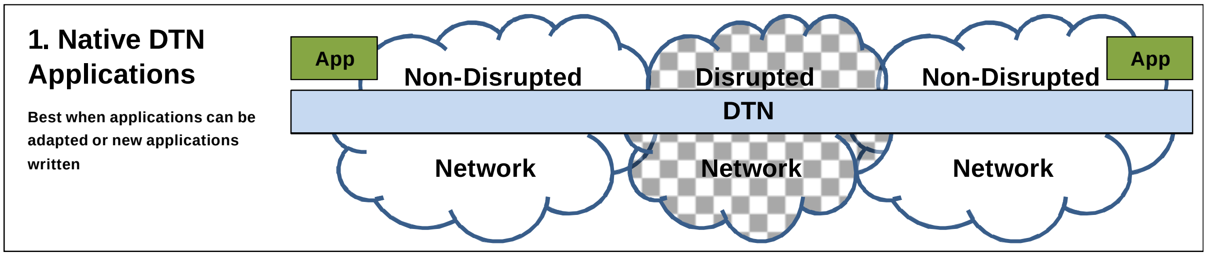

1. Native DTN Applications
Best when applications can be adapted or new applications written
App
Non-Disrupted
Disrupted
Non-Disrupted
App
DTN
Network
Network
Network

**Figure C-2: In-Situ Network Support: DTN End-to-End**

## C6.3 APPLICATION LAYER GATEWAYS

In the second method, some node, either a member of the in-situ group or an infrastructure node such as a relay orbiter, serves as a gateway between the local and long-haul communications. This gateway has to terminate the local networking protocols and pass the data to a peer using the Bundle Protocols. For such a gateway to function, it is likely that it would have to know the specifics of the Application layer protocol involved. For example, if the application periodically issued a heartbeat timer and expected a response, the gateway could provide that response in order to maintain the illusion of connectivity and prevent the application from terminating communications. This is somewhat problematic, as the application may make other decisions based on its connectivity state, and the gateway is masking any disconnectivity on the long-haul network.

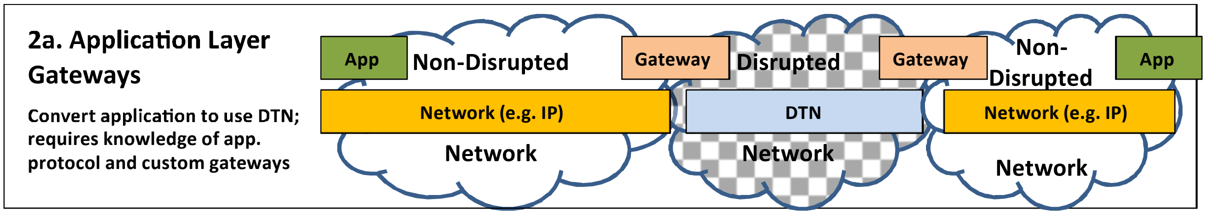

2a. Application Layer Gateways
Convert application to use DTN; requires knowledge of app. protocol and custom gateways
App
Non-Disrupted
Gateway
Disrupted
Gateway
Non-Disrupted
App
Network (e.g. IP)
DTN
Network (e.g. IP)
Network
Network
Network

**Figure C-3: In-Situ Network Support: Application Layer Gateways**

An asymmetric version of this approach is also possible, where one side of the communication simply uses Bundle Protocols while the other uses a gateway to translate between DTN and another protocol suite.

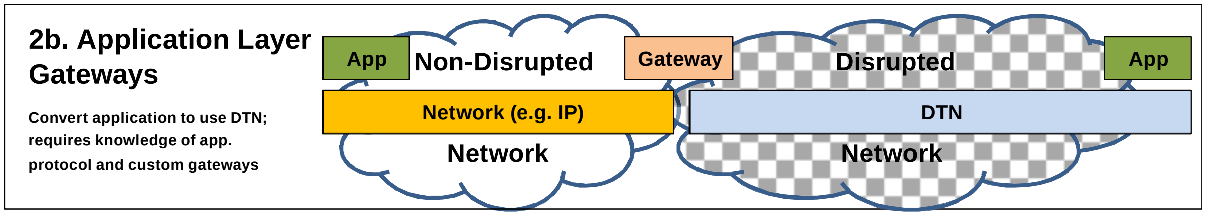

2b. Application Layer Gateways
Convert application to use DTN; requires knowledge of app. protocol and custom gateways
App
Non-Disrupted
Gateway
Disrupted
App
Network (e.g. IP)
DTN
Network
Network

**Figure C-4: In-Situ Network Support: Asymmetric Application Layer Gateways**

## C6.4 TUNNELING IN-SITU NETWORK PROTOCOLS OVER DTN

In the third approach, PDUs from the in-situ communications protocol are tunneled across the long-haul network inside DTN PDUs. If the Internet suite is used as the in-situ protocol suite, this *might* work for UDP packets but will probably not work for TCP. The reason is that TCP requires frequent acknowledgements from the receiver that data is in fact reaching it, and if there is significant delay (either very long one-way light times or storage delay due to temporary network disruptions) TCP will cease to function.

In order for a tunneled approach to work using the Internet suite as the in-situ networking protocol suite, the following issues would need to be addressed:

a) Name Resolution from the In-Situ Network to Nodes across the Long-Haul Link. Typically IP communications use Domain Name System (DNS) names (e.g., www.ccsds.org) as an indirection mechanism instead of directly using IP addresses such as 66.192.184.81 This allows service providers to move services easily by changing only the service entry in the DNS. The problem here is that resolving a DNS name to an IP address must take place before any communications, and that resolution involves communication with a DNS server that knows the mapping. For the case of a remotely-deployed in-situ piece of IP infrastructure, it is likely that a DNS request would have to traverse the disrupted network, and no communications could take place until the resolution completed. Alternatively, a local DNS cache could be deployed together with the in-situ network, but keeping the cache mappings up-to-date could be bandwidth-intensive, especially if deployed nodes wanted to be able to reach a large number of terrestrial endpoints.

   If DNS is not used, host tables (static mappings of DNS names to IP addresses) could be used, or IP addresses could be used directly by the in-situ devices. Both of these mechanisms are brittle, as they constrain the terrestrial operators not to change the IP addresses of services.

b) Static IP Routes over Long-Haul Link. IP routes would have to be in place to route IP packets correctly over the long-haul portion of the network. Since no dynamic routing protocol for IP works well in highly delayed and/or disrupted environments, these would presumably be static routes.

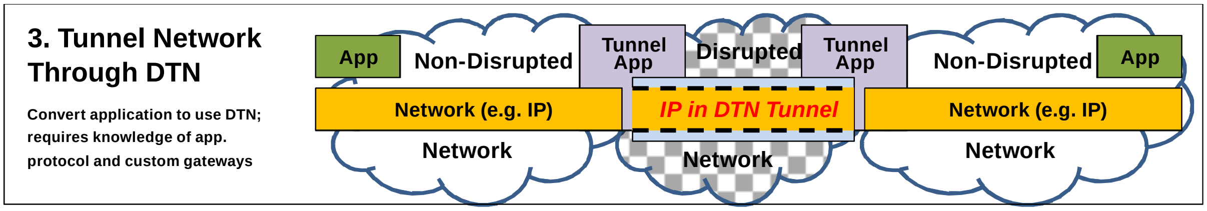


**Figure C-5: In-Situ Network Support: Tunneling over DTN**

### C6.5 CONCLUSIONS

This document does not dictate which of the above methods are used to connect in-situ networks with each other and/or with the Earth. In each case above, the endpoints of communications as viewed by the Bundle Protocol are applications, whether they are the user applications, Application layer gateways, or tunnel endpoints. It is expected that not all applications may be amenable to the Application layer gateway approach described above, as some applications may require true end-to-end communications that cannot be gatewayed, such as authentication traffic. There are certainly some IP-based applications that are not amenable to the tunneling approach, as they have embedded Application layer timers that will not function if the tunneled IP packets are stored for significant periods of time (or even if the light-time delays in the Disrupted Network are large enough).

# ANNEX D

# COMPARISON OF CANDIDATE SPACE INTERNETWORKING PROTOCOLS

Table D-1 contains the comparison matrix of proposed protocol alternatives against the requirements listed in section 4 of the document. Custom data forwarding is not included in the table. While custom data forwarding solutions at each node *could* in principle support all of the requirements, the interfaces required to support the services end-to-end would be prohibitively difficult to operate.

**Table D-1: Comparison of Proposed Approaches against Requirements for a Space Internetworking Protocol**

| DTN GB Requirement | CFDP SFO | CFDP EP | Internet Protocols | DTN Protocols |
|---|---|---|---|---|
| 4.2.2.1.1 Communications shall be supported to a spacecraft via zero or more intermediate relays. | Supported | Supported | Supported | Supported |
| 4.2.2.1.2 It shall be possible to use local, in-situ networking technologies different from the end-to-end space internetwork technology. | Supported (with gateway between ‘global network’ and in-situ technology) | Supported (with gateway between ‘global network’ and in-situ technology) | Supported (with gateway between ‘global network’ and in-situ technology) | Supported (with gateway between ‘global network’ and in-situ technology) |
| 4.2.2.1.3 The system shall support a general class of applications, including at least file transfer and messaging. | Supported (messaging supported via ‘message-to-user’ capability of CFDP and limited to 255 bytes per message). | Supported (messaging supported via ‘message-to-user’ capability of CFDP and limited to 255 bytes per message). | Supported (e.g., UDP) | Supported |
| 4.2.2.1.4 Management information relating to data transfer shall be collected in all nodes. | Possible some development required. | Possible some development required. | Possible; standard protocols and applications for network management exist. | Possible some development required. |
| 4.2.2.1.5 Management information relating to data transfer shall be made available to network operators. | Possible some development required. | Possible some development required. | Possible some development required. | Possible some development required. |
| 4.2.2.1.7 It shall be possible to configure routing to automatically fail over to redundant routes if such routes are available. | Possible routing mechanisms would need to be developed. | Possible routing mechanisms would need to be developed. | Supported via dynamic routing protocols. | Supported via dynamic routing and/or forwarding (e.g., CGR). |

| DTN GB Requirement | CFDP SFO | CFDP EP | Internet Protocols | DTN Protocols |
|---|---|---|---|---|
| 4.2.2.1.8 Communications firewalls shall be implemented at interoperability points to guarantee mission security. | Possible | Possible | Possible | Possible |
| 4.2.2.1.9 Methods for user authentication shall be incorporated with authenticated users having associated levels of permission and resource allocation. | Possible | Possible | Possible | Possible |
| 4.2.2.1.10 Data privacy between users shall be provided. | Possible | Possible | Possible | Possible |
| 4.2.2.1.11 It shall be possible to use multiple ground stations to communicate with a single space asset with some ground stations providing downlink capability only. | Possible | Possible | Possible; would probably require static/managed routing (standard routing protocols do not support partitioned networks). Reliable data communications would be difficult (standard protocols do not support partitioned networks). | Possible |
| 4.2.2.1.12 It shall be possible to route data from the ground station directly to destinations without routing via the control center. | Possible | Possible | Possible | Possible |
| 4.2.2.1.13 It shall be possible to implement Application layer firewalls at interoperability points to guarantee mission safety. | Possible | Possible | Possible | Possible |
| 4.2.2.1.14 ‘Hardware commanding’ of spacecraft by embedding special command sequences in either frames or packets shall be supported. | Possible; requires special hardware commanding application. | Possible; requires special hardware commanding application. | Possible; requires special hardware commanding application. | Possible; requires special hardware commanding application. |
| 4.2.2.2.1 It shall be possible to send a file to an application on board a spacecraft that can, either by autonomous methods or managed by mission / infrastructure management or a combination of both, convey the file to a second spacecraft. | Possible; development required. | Possible; development required. | Possible; development required. | Possible; development required. |

| DTN GB Requirement | CFDP SFO | CFDP EP | Internet Protocols | DTN Protocols |
|---|---|---|---|---|
| 4.2.2.2.2 The end-to-end infrastructure and protocols shall be capable of transferring, as Service Data Units (SDUs), the Protocol Data Units (PDUs) of the following CCSDS protocols: CCSDS File Delivery Protocol (CFDP), Space Packet Protocol (SPP), Encapsulation Packet Protocol (EP), Telemetry (TM), Telecommand (TC), and Asynchronous Messaging System (AMS). | Possible | Possible | Possible | Possible |
| 4.2.2.2.3 The end-to-end infrastructure and protocols shall provide the services specified as required of the underlying layers of the CFDP, SPP, EP, Telemetry, Telecommand, and AMS protocols. | Yes | Yes | Yes | Yes |
| 4.2.2.2.4 The end-to-end infrastructure and protocols shall be capable, under the direction of users and/or mission/infrastructure network management, of supporting qualities of service with respect to data completeness. | Supported. | Supported. | Possible; development would be required to support data completeness in the case of half-duplex connections. More development required to support temporarily partitioned networks. | Supported. |
| 4.2.2.2.5 The end-to-end infrastructure and protocols shall be capable, under the direction of users and/or mission/infrastructure network management, of supporting qualities of service with respect to data errors. | Supported. (CFDP FDUs have checksums and PDUs have optional CRCs.) | Supported. (CFDP FDUs have checksums and PDUs have optional CRCs.) | Supported (UDP datagrams may or may not contain a checksum). | Supported (sources may or may not request payload integrity). |
| 4.2.2.2.6 The end-to-end infrastructure and protocols shall be capable, under the direction of mission/infrastructure network management, of supporting qualities of service with respect to data sequencing (depends on tolerance to out of sequence PDUs of upper layer protocols). | Sequenced delivery of streams can be supported; development required. | Sequenced delivery of streams can be supported; development required. | Sequencing provided by end systems (TCP) or applications (UDP). | Sequenced delivery of streams can be supported; development required. |

| DTN GB Requirement | CFDP SFO | CFDP EP | Internet Protocols | DTN Protocols |
|---|---|---|---|---|
| 4.2.2.2.7 The end-to-end infrastructure and protocols shall be capable, under the direction of the application and mission/infrastructure network management, of supporting QoS with respect to data priority. | Potentially somewhat supported (CFDP FDUs have flow labels, but they only prioritize at file granularity; no prioritization of PDUs.) | Potentially somewhat supported (CFDP FDUs have flow labels, but they only prioritize at file granularity; no prioritization of PDUs.) | Supported (differentiated services). | Supported (Bundle priority and extended class-of-service block). |
| 4.2.2.2.8 The end-to-end infrastructure and protocols shall be capable, under the direction of users and/or mission/infrastructure network management, of supporting qualities of service with respect to data availability (via e.g., alternate routes). | Possible; development required. | Possible; development required. | Supported via automated routing protocols and/or route failover. | Supported (Contact Graph Routing) |
| 4.2.2.2.9 The Space Internetworking Protocols (e.g., BP and IP) shall be capable of operating over the CCSDS Encapsulation Protocol. | | | Supported. | Supported. |
| 4.2.2.3.1 The transfer protocols shall be capable of transferring application data units completely (reliably) when required by applications. If an application does not require complete delivery, the transfer protocols may deliver incomplete data (data with holes). | Supported. | Supported. | Supported (e.g., UDP) | Supported (via unreliable Bundle delivery). |
| 4.2.2.3.2 The transfer protocols shall be capable of transferring complete sequences of messages. | Supported (reliable delivery of each message). | Supported (reliable delivery of each message). | Supported (via TCP, for example.) | Supported (via custody transfer of each message). |
| 4.2.2.3.3 The transfer protocols shall be capable of transferring sequences of messages in-sequence. | Metadata PDU can contain multiple brief (up to 255 bytes) messages. Additional development would be required to support sequenced file delivery. | Metadata PDU can contain multiple brief (up to 255 bytes) messages. Additional development would be required to support sequenced file delivery. | Sequencing provided by end systems (TCP) or applications (UDP). | Sequenced delivery of streams can be supported; development required. |
| 4.2.2.3.4 It shall be possible to transfer a file over a disrupted link, retaining the state of the file transfer between contact periods. | Supported. | Supported. | Not supported; some end-to-end recovery mechanisms (e.g., download managers) exist. | Supported (via LTP, for example.) |

| DTN GB Requirement | CFDP SFO | CFDP EP | Internet Protocols | DTN Protocols |
|---|---|---|---|---|
| 4.2.2.3.5 It shall be possible to 'hand-over' the transmission of a file from one intermediate hop to another (e.g., transmission starts using ground station A, A looses visibility and hands-over to ground station B). | Possible by changing routes (managed or automatic). | Possible by changing routes (managed or automatic). | Possible by changing routes (managed or automatic). | Possible by changing routes (managed or automatic). |
| 4.2.2.3.6 Data transfer shall be capable of operating over simplex links (with limited QoS). | Supported; reliable delivery across such links supported provided there is eventual return connectivity. | Supported; reliable delivery across such links supported provided there is eventual return connectivity. | Supported (UDP). | Supported; reliable delivery across such links supported provided there is eventual return connectivity. |
| 4.2.2.3.7 Data transfer shall be capable of operating over network paths with widely differing capacities (up to 10,000:1) | Potentially supported. Reliable transmission is possible only when files are large enough to ensure acknowledgment traffic rate does not exceed limit. | Potentially supported. Reliable transmission is possible only when files are large enough to ensure acknowledgment traffic rate does not exceed limit. | Partially supported (UDP). No reliable transmission over these links. | Supported by LTP. |
| 4.2.2.3.8 Data Transfer protocols shall be independent of application data content. | Yes | Yes | Yes | Yes |
| 4.2.2.3.9 File transfer may be initiated by the sender of a file, the receiver of a file or a third party. | Yes | Yes | Yes | Yes |
| 4.2.2.3.10 File transfer shall take place between file stores under the control of file service user entities. | Yes | Yes | Yes | Yes |
| 4.2.2.3.11 Message transfer shall take place between message service user entities. | Yes | Yes | Yes | Yes |
| 4.2.2.3.12 Data transfer shall be possible over multiple concatenated heterogeneous data Transport layers. | Yes | Yes | Yes | Yes |
| 4.2.2.3.13 Given suitable QoS attributes when data is submitted and suitable network connectivity, it shall be possible to verify completeness of the data transfer and to notify the data transfer originator about this. This shall be possible regardless of other QoS attributes (e.g., completeness). | Supported (for single files). Support for notification of sequences of files would require development. | Supported (for single files). Support for notification of sequences of files would require development. | Supported (TCP). | Supported (for single Bundles). Support for notification of sequences of Bundles would require development. |

| DTN GB Requirement | CFDP SFO | CFDP EP | Internet Protocols | DTN Protocols |
|---|---|---|---|---|
| 4.2.2.3.14 Data transfer shall support priority and preemption mechanisms in all nodes. | Possible; development required. | Possible; development required. | Possible; development required. | CFDP/ION fully supports prioritization and supports preemption of large file transfers (because each PDU is a Bundle and prioritization is at Bundle granularity). |
| 4.2.2.3.15 It shall be possible to transfer file metadata as part of the file transfer protocol or using a messaging protocol. | Supported. | Supported. | N/A | N/A |
| 4.2.2.3.16 Data transfer protocols shall not require simultaneous availability of the communication link between all nodes involved in the data delivery/routing. | Supported. | Supported. | NOT supported. | Supported. |
| 4.2.2.3.17 It shall be possible to use the same data transfer protocol in the Ground-to-Space link, in the Space-to-Space link and between ground nodes (Ground-to-Ground). | Supported. | Supported. | Partially supported (with caveats when implementing IP over simplex and half-duplex links and networks with temporary partitioning). | Supported. |
| 4.2.2.3.18 Data retransmission strategy shall be flexible to allow opportunistic (automated) retransmission of data when links become available while still respecting quality of service conditions. | Supported. | Supported. | Partially supported (TCP). | Supported. |
| 4.2.2.3.19 Retransmitted data shall, by default, assume the same priority as the original data. | Supported. | Supported. | Supported. | Supported. |
| 4.2.2.3.21 It shall be possible to demultiplex the SDUs contained in Network layer PDUs to specific upper-layer entities. | There is only one CFDP user; demultiplexing would need to be a function of the filename sent. | There is only one CFDP user; demultiplexing would need to be a function of the filename sent. | Supported. | Supported. |

| DTN GB Requirement | CFDP SFO | CFDP EP | Internet Protocols | DTN Protocols |
|---|---|---|---|---|
| 4.2.2.3.22 The data transfer protocols shall be able to operate in a communications environment characterized by large transmission delays. | Supported. | Supported. | Partially supported (UDP is insensitive to delays); large delays would limit the use of some routing protocols and could impact performance of IP mechanisms (if used) such as name-to-address resolution and reliable data delivery. | Supported. |
| 4.2.2.3.23 The data transfer protocols shall be able to operate in a communications environment characterized by unreliable, noisy communication links. | Supported. | Supported. | Supported. | Supported. |
| 4.2.2.3.24 The data transfer protocols shall be able to operate in a communications environment characterized by interrupted visibility between communication nodes due to predictable causes (e.g., orbital visibility) | Supported. | Supported. | NOT supported. | Supported. |
| 4.2.2.3.25 The data transfer protocols shall be able to operate in a communications environment characterized by unpredictable disruptions due to failures. | Supported. | Supported. | NOT supported. | Supported. |
| 4.2.2.3.26 The protocol shall have a mechanism for carrying a priority field that may be affected by the user and/or management/policy at the sending node. | Supported. | Supported. | Supported. | Supported. |
| 4.2.3.27 Management / policy at intermediate nodes (nodes other than the source) may override the priority treatment indicated in the priority field of a space internetworking PDU. | Possible; development required. | Possible; development required. | Supported (DSCP and DSCP modification). | Possible; development required. |
| 4.2.2.3.28 It shall be possible for the file transfer protocol to perform multiple file transfer transactions in parallel (e.g., in order to initiate the delivery of file 'n+1' before receiving confirmation of successful transfer of file 'n'). This is essential in order to optimize the use of the available bandwidth. | Supported. | Supported. | N/A | N/A |
| 4.2.2.4.1 It shall be possible to observe the progress of data transfers by local or remote data management entities. | Possible; development required. | Possible; development required. | Possible; development required. | Possible; development required. |

| DTN GB Requirement | CFDP SFO | CFDP EP | Internet Protocols | DTN Protocols |
|---|---|---|---|---|
| 4.2.2.4.2 It shall be possible to observe the state of data transfer queues (file or message) by local or remote data management entities. | Possible; development required. | Possible; development required. | Typically not possible (queues are extremely transient in IP). | Possible; development required. |
| 4.2.2.4.3 It shall be possible to control data transfer queues by reordering, deleting, suspending/resuming transmission of queued items by local or remote data management entities. | Possible; development required. | Possible; development required. | Possible; development required. | Possible; development required. |
| 4.2.2.4.4 It shall be possible to control the actions of file transfer applications with respect to stop (cancel), suspend and resume (global or individual files) by local or remote data management entities. | Possible; development required. | Possible; development required. | N/A | N/A |
| 4.2.2.4.5 It shall be possible to preempt data transfers either locally to the sending entity or remotely from a remote manager. | Possible; development required. | Possible; development required. | Possible; development required. | Possible; development required. |
| 4.2.2.4.6 Suspension and resumption of transfer at transmitting or receiving ends may be initiated by a local management entity in response to an anticipated or unanticipated outage. [This is possible wherever the file transfer application is transmitting the file.] This is a requirement on CFDP or the CFDP user. | Supported. | Supported. | N/A | N/A |
| 4.2.2.4.7 It shall be possible to establish primary and backup routes through the end-to-end data path at a network planning facility and to distribute this information to the nodes concerned. | Possible; development required. | Possible; development required. | Supported. | Possible; development required. |
| 4.2.2.4.8 Synchronization of route changes must be managed in the end-to-end network. | Possible | Possible | Possible | Possible |
| 4.2.2.4.9 It shall be possible to terminate data transmission via a relay node A, delete the data buffered at A, and resume data transmission via another next-hop relay, if necessary. | Possible; requires development of network management capabilities. | Possible; requires development of network management capabilities. | Possible; requires development of network management capabilities. | Possible; requires development of network management capabilities. |

| DTN GB Requirement | CFDP SFO | CFDP EP | Internet Protocols | DTN Protocols |
|---|---|---|---|---|
| 4.2.2.4.10 The data transfer protocols shall provide to the destination the time of transmission and receipt of the application data unit being delivered. | There is no time tag in CFDP (though it could be added as a non-standard Metadata TLV). | There is no time tag in CFDP (though it could be added as a non-standard Metadata TLV). | Supported (IP timestamps for sending time; implementation issue / some development required at receiver to indicate time of receipt). | Supported (Bundle creation time is part of the Bundle Protocol; implementation issue / some development required at receiver to indicate time of receipt). |
| | | | | |
| 4.2.2.5.1 Application layer content (e.g., files, messages) for onward transmission to a spacecraft may be examined and checked for mission critical effects at a mission control entity and blocked if necessary. | Possible; development of Application layer firewall required. | Possible; development of Application layer firewall required. | Possible; development of Application layer firewall required. | Possible; development of Application layer firewall required. |
| 4.2.2.5.3 An application on the last hop relay node may extract TCs from an immediate or delayed TC file and radiate them as TCs to their destination (typically orbiter to lander). | Possible; development of TC relay application required. | Possible; development of TC relay application required. | Possible; development of TC relay application required. | Possible; development of TC relay application required. |
| 4.2.2.5.4 An application on the first hop relay node may assemble TM packets received from another entity and assemble them into a TM file for further transmission. | Possible; development of TM relay application required. | Possible; development of TM relay application required. | Possible; development of TM relay application required. | Possible; development of TM relay application required. |

# ANNEX E

# DTN APPLIED TO SCENARIOS

## E1 OVERVIEW

This annex describes how the DTN Protocol Suite, and particularly the Bundle Protocol, could be used to support a number of simple scenarios. Special attention is paid in the beginning to naming and routing considerations in E3 that describes a scenario with a lander operations center, an orbiter operations center, and a spacecraft. Issues related to multiple ground stations are addressed in E4, while E5 discusses a scenario with multiple orbiters communicating with a landed element on another planet.

## E2 ASSUMPTIONS

It is assumed that there is a single Bundle Protocol engine at each node in each figure unless otherwise noted. In particular, spacecraft buses and payloads share a single Bundle Protocol agent on each spacecraft unless otherwise noted.

It is also assumed that the nominal Earth-to-space data link rate is 20kb/s, the nominal space-to-Earth data link rate is 100kb/s, and the nominal terrestrial data rate is 5Mb/s for concreteness of the examples. Two cases for the location of the terminus of the space link are considered: either at the orbiter control center or at the ground station.

The following assumptions regarding node number allocation apply throughout:

| Node Numbers | Location |
|---|---|
| 400 | Ground Station 1 |
| 500 | Ground Station 2 |
| 600 | Orbiter Control Center (OCC) |
| 700 | Payload Control Center (PCC) |
| 900-1000 | Unassigned |
| 1100 | Orbiter 1 |
| 800 | Orbiter 2 |
| 1200 | Orbiter 3 |
| 1300 | Landed Element (Gateway element for in-situ network) |

The Endpoint Identifier node numbers for Ground Station 2 and Orbiter 2 were intentionally chosen to be non-contiguous with other similar elements to highlight the mechanisms needed to accommodate such situations.

# E3 SINGLE-SPACECRAFT WITH ONE GROUND STATION

## E3.1 GENERAL

The simplest scenario is a single control center communicating with a spacecraft via a single ground station, as shown in figure E-1.

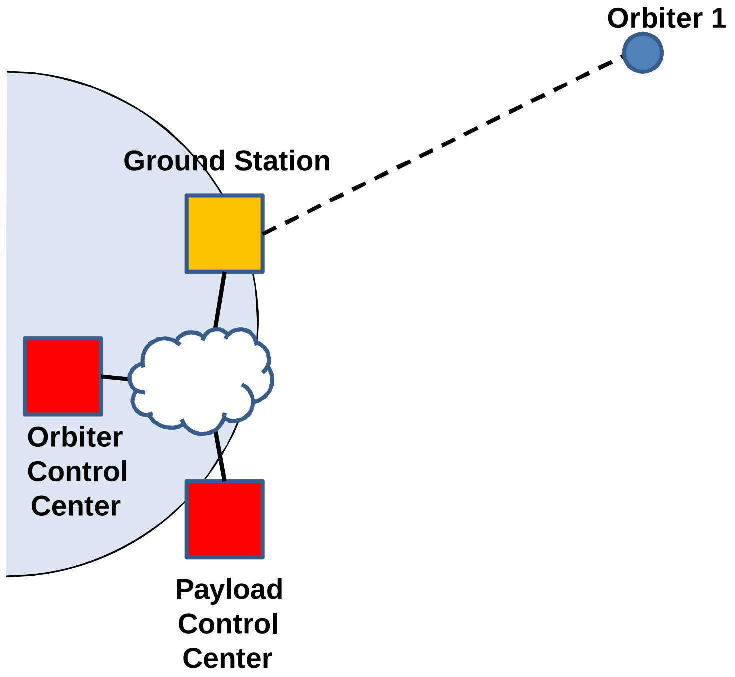


**Figure E-1: Single Spacecraft, Single Ground Station Scenario**

## E3.2 IF THE GROUND STATION DOES NOT IMPLEMENT A BP ROUTER

### E3.2.1 General

In this example the Ground Station does not implement a Bundle Protocol router. Considered is the case where policy requires all traffic from the Payload Control Center (PCC) to be inspected by the Orbiter Control Center before being sent to the spacecraft, and that this policy inspection is implemented at the Application layer. That is, traffic from the PCC destined for the orbiter is transmitted to a BP-aware application in the orbiter control center.

It is also assumed that the space link terminates in the Orbiter Control Center (OCC) so that direct communication between the spacecraft and the PCC (without going through the OCC) is not possible. It is assumed CCSDS data links (some combination of TC/TM/AOS/Prox-1) is employed between the OCC and the orbiter, using CSTS to tunnel the space link across the terrestrial Internet between the OCC and the ground station.

### E3.2.2 Protocol Stack Diagram

Figure E-2 shows a protocol stack diagram for this configuration, where the details of the Data Link layer are intentionally omitted. While the OCC shows two separate DTN stacks leading into the application, in practice a single DTN instance would suffice. Two separate instances are shown to emphasize that data from the PCC is processed, at the Application layer, inside the orbiter control center before being forwarded to the spacecraft.

Figure E-2 also shows the use of LTP underneath DTN. If the space data link were reliable (AOS in reliable mode, for example), then DTN could be run directly on top of Encapsulation Packets without the use of LTP.

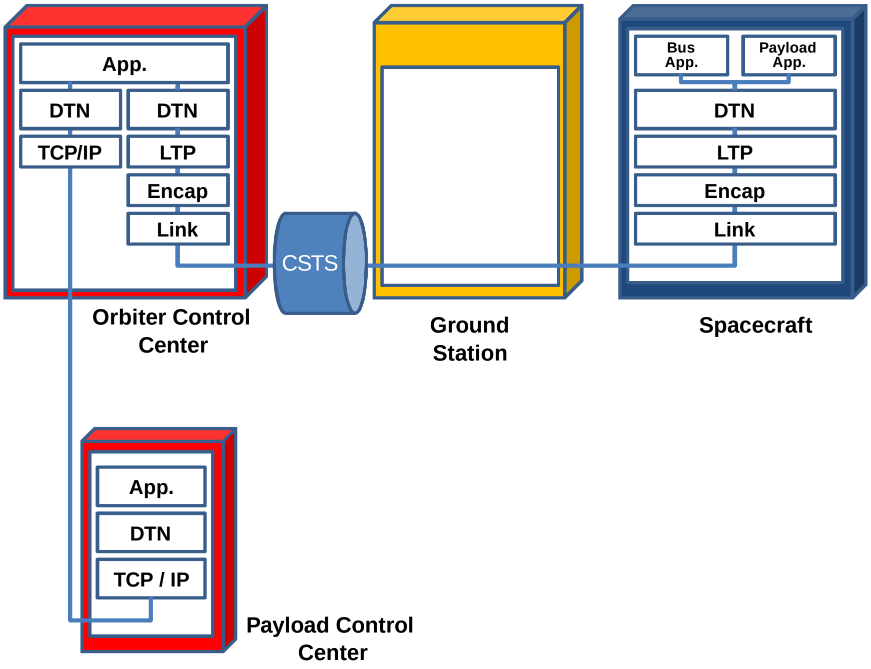


**Figure E-2: Protocol Stack Diagram**

### E3.2.3 Contact Information

Some mechanism for determining routes needs to be established. Contact Graph Routing can be thought of as a time-aware forwarding process that takes as inputs the contact times between various nodes and then, when presented with a message to forward at a particular time *t*, attempts to find a path to the destination starting at *t* that meets the various links' connectivity and transmission rate constraints.

As an example, the topology shown above may be considered with the following assumptions about link connectivity:

Terrestrial contacts:

Always available (Ground stations, control centers, and other Earth-based nodes have continuous connectivity among themselves).

Ground Station-Orbiter 1 contacts:

600-1200s;

1500-2100s.

### E3.2.4 Contact Graph Routing Information in Payload Control Center

#### E3.2.4.1 General

| Source | Destination | Start (s) | Stop (s) | Bandwidth (b/s) | Notes |
|---|---|---|---|---|---|
| PCC | OCC | 0 | 900 | 10K | The PCC may contact information that does not necessarily correspond to the 'true' contacts. This provides a mechanism to throttle the PCC-to-orbiter traffic so that it does not either consume the space link or build up at the OCC waiting to be transmitted. |
| OCC | Orbiter 1 | 600 | 900 | 10K | |
| PCC | OCC | 1200 | 1500 | 10K | |
| OCC | Orbiter 1 | 1800 | 2100 | 10K | |

#### E3.2.4.2 Contact Information for Payload Control Center:

By not using the full contact periods and Earth-to-space data rates at the PCC, the amount of data the PCC attempts to push through the OCC can be controlled. In the first pair of contacts above, the PCC is allowed to communicate with the orbiter during the first half of the pass and is limited to 10 kb/s during that time. In the second pair of contacts, the PCC is required to transmit data to the OCC in advance of the second pass, and the data is emitted to the orbiter only during the second half of the pass.

This provides a mechanism to provide backpressure to the PCC, preventing it from simply shoving all of its data at the OCC and building a possibly large queue there.

#### E3.2.4.3 Routing Information for Payload Control Center

| Destination(s) | Next Hop |
|---|---|
| OCC | OCC, TCP connection |

There is no route in the PCC's forwarding table from the PCC to the orbiter. The only destination for PCC traffic is the OCC. This is consistent with the assumption that PCC traffic is inspected by a BP-aware Application layer entity in the OCC.

A possible alternate mechanism to ensure that payload commands did not adversely affect the orbiter would be to simply route all traffic to the orbiter via the OCC and to have a BP-aware firewall process in the OCC. Such a process could inspect Bundles and ensure that no Bundles destined for the orbiter bus had source EIDs in the PCC. Such a capability would be similar to the 'iptables' capability in Linux, for example, and does not currently exist for the Bundle Protocol. In this case, the PCC routing table would contain an entry for the orbiter, with a next hop of the OCC.

### E3.2.5 Contact Graph Routing Information in the Orbiter Control Center

#### E3.2.5.1 Contact Information for Orbiter Control Center

| Source | Destination | Start (s) | Stop (s) | Bandwidth (b/s) | Notes |
|---|---|---|---|---|---|
| OCC | PCC | 0 | N/A | 5M | OCC constantly connected to PCC. |
| OCC | Orbiter 1 | 600 | 1200 | 20K | A 10-minute pass between the OCC and the orbiter (the ground station is not a DTN node and so is transparent to DTN routing). |
| OCC | Orbiter 1 | 1500 | 2100 | 20K | Another 10-minute pass between the OCC and the orbiter. |

#### E3.2.5.2 Routing Information for Orbiter Control Center

| Destination(s) | Next Hop |
|---|---|
| Orbiter 1 | Orbiter 1, CCSDS space link via GS using CSTS |
| PCC | PCC, TCP connection |

### E3.2.6 Contact Graph Routing Information on Orbiter 1 With No Spacecraft-to-PCC Routing

#### E3.2.6.1 General

It is assumed that there is no underlying internetworking layer shared among the spacecraft and the PCC. In this case, all telemetry from the orbiter must first go to the OCC.

#### E3.2.6.2 Contact Information for Orbiter

| Source | Destination | Start (s) | Stop (s) | Bandwidth (b/s) | Notes |
|---|---|---|---|---|---|
| OCC | PCC | 0 | N/A | 5M | OCC constantly connected to PCC |
| Orbiter 1 | OCC | 600 | 1200 | 100K | |
| Orbiter 1 | OCC | 1500 | 2100 | 100K | |

#### E3.2.6.3 Routing Information for Orbiter 1

| Destination(s) | Next Hop |
|---|---|
| ALL | OCC, CCSDS space link |

In this configuration where the payload and bus share a single DTN router, the only mechanism for limiting the amount of traffic the payload offers to the space-to-Earth link is via priority. That is, if there is bus traffic with higher priority, that traffic will be forwarded before the lower-priority payload traffic. If the payload and bus had separate DTN router instances, or if there were an underlying internetwork making the PCC directly reachable from the orbiter, then the amount of payload traffic could be controlled via the contact information supplied to the payload.

### E3.2.7 Contact Graph Routing Information on Orbiter 1 With Spacecraft-to-PCC Routing

#### E3.2.7.1 General

If there were a common underlying internetworking layer (e.g., IP) connecting the spacecraft to the various ground nodes and if the space link terminated at the ground station, it would be possible to route data directly from the ground station to the payload control center.

#### E3.2.7.2 Contact Information for Orbiter

| Source | Destination | Start (s) | Stop (s) | Bandwidth (b/s) | Notes |
|---|---|---|---|---|---|
| OCC | PCC | 0 | N/A | 5M | OCC constantly connected to PCC |
| Orbiter 1 | OCC | 600 | 1200 | 100K | |
| Orbiter 1 | OCC | 1500 | 2100 | 100K | |
| Orbiter 1 | PCC | 600 | 900 | 50K | This allows restricting the amount of telemetry data destined for the PCC by restricting the time and/or bandwidth of the Orbiter-to-PCC contacts. |
| Orbiter 1 | PCC | 1000 | 1100 | 20K | |

In this case, the amount of traffic the payload offers to the space-to-Earth link is limited by the bandwidths of the Orbiter 1-to-PCC contacts in the payload contact information. This merely puts a cap on the payload bandwidth; the bus is still allowed to communicate with the OCC using the full 100 kb/s bandwidth. In the case that the bus uses the full 100 kb/s and there is concurrent payload traffic; the order in which Bundles are placed onto the space-to-Earth link would be determined by the relative priorities of the Bundles.

### E3.2.8 Routing information for Orbiter 1

| Destination(s) | Next Hop |
|---|---|
| OCC | OCC, IP-based convergence layer over CCSDS space link |
| PCC | PCC, IP-based convergence layer over CCSDS space link |

## E3.3 IF THE GROUND STATION IMPLEMENTS A BP ROUTER

### E3.3.1 General

The basic structures above do not change if the ground station implements a BP router. In this case the ground station BP router would appear in the contact and routing information as an intermediate hop, much as the OCC appears as an intermediate hop to the PCC.

Operationally, a difference between the two scenarios would be that, if the ground station implemented a BP router, the OCC would need to decide what contact information to use for the OCC-to-ground station link. If the OCC saw the OCC-to-ground station link as continually available with high bandwidth, the OCC would be able to queue large amounts of data at the ground station. If this is intended, it might improve performance by allowing the ground station to precompute frames for transmission, etc. Building a large unintended queue at the ground station could lead to large round-trip times and unnecessary retransmissions of data from the OCC.

### E3.3.2 Protocol Stack Diagrams

If the ground station implements a DTN router, then communications between the OCC and the ground station can use the TCP/IP protocol suite, with the ground station DTN implementation converting to the use of a space-compatible stack.

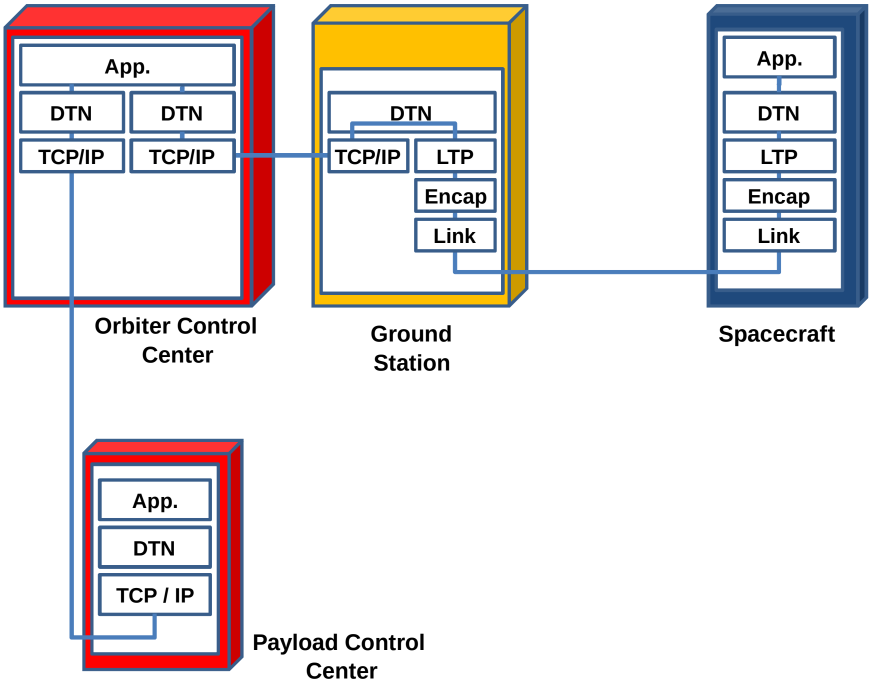


**Figure E-3: Protocol Stack Diagram Where the Ground Station Implements a DTN Router**

## E3.4 ISSUES

### E3.4.1 Knowledge of Transmission Opportunities

It is assumed here that the transmission opportunities to the orbiter are known in advance and that some sort of service management mechanism is used to configure the ground station. Similarly, the connectivity information between the ground station and the orbiter is assumed to be provided to the OCC so that it can be used to derive the types of contact and routing information discussed above.

#### E3.4.2 Transmission When Not Connected

What happens when a terrestrial node attempts to transmit a Bundle to the orbiter when the ground station is not connected to the orbiter depends on whether or not there is another Bundle Protocol router between the data source and the orbiter. For the case where the ground station does not implement a Bundle router, if the PCC attempts to transmit when the ground station is not communicating with the orbiter, Bundles will be stored at the OCC until connectivity resumes. When connectivity resumes, the prioritization process at the OCC will choose the order in which PCC and OCC Bundles are emitted towards the spacecraft.

If the ground station does implement a Bundle router, then either the PCC or the OCC may emit Bundles destined for the orbiter at any time, and the ground station will queue those Bundles for transmission until the next opportunity.

#### E3.4.3 Effects of Simplex Connectivity between the Orbiter and Ground Station

The Bundle Protocol itself does not depend on contemporaneous bi-directional communication between Bundle routers. Indeed, if communications were entirely simplex and if the underlying communication mechanism supported simplex communications, the Bundle Protocol could still function, though its positive-acknowledgement-based reliability mechanism (custody transfer) would be compromised.

#### E3.4.4 Paths for Downlinked Data

If the ground station implements a Bundle Protocol router, then it can be a branching point for data coming down from the orbiter. In particular, data for the PCC may be forwarded directly from the ground station to the PCC without going through the OCC.

#### E3.4.5 Effects of Unexpected Changes in Connectivity

The exact mechanism for recovery from unexpected loss of connectivity depends on the capabilities of the BP stack implementation (including the Bundle Protocol implementation and the capabilities of the underlying (convergence layer) mechanisms):

- Reactive fragmentation: if the transmitting BP router knows how much data from a particular Bundle transmission has been successfully received when contact with the orbiter is lost, it could choose to *reactively fragment* the Bundle. This assumes that the receiver forms a Bundle fragment from the data already received, and the sender forms a fragment from the last known byte received. The fragment at the source is then re-routed, possibly via another path if one is available.
- Link-layer persistence: The Licklider Transmission Protocol can use bidirectional communications to detect link outage and suspend operations when connectivity is lost and resume transmission when connectivity is restored. If LTP is used as the underlying mechanism for communications, then this mechanism can be invoked.

– Abort and retransmit: The sender may simply abort transmission of the Bundle and retransmit it at the next opportunity.

#### E3.4.6 Protection of the Spacecraft from Adverse Payload Commands

Described above are two ways in which the OCC could ensure that data sent from entities other than the OCC were precluded from adversely affecting the Orbiter Spacecraft itself. In the first of these, all traffic destined for the orbiter originating outside the OCC must be transmitted to a particular BP-aware application in the OCC for inspection and forwarding. In this case, some other metadata might be required to identify the final destination (if one OCC were servicing multiple orbiters, for example). The Bundle Protocol contains mechanisms for attaching arbitrary metadata to Bundles that could be used for this purpose.

A second mechanism to ensure that PCC Bundles did not adversely affect the orbiter would be to route all such Bundles through the OCC and to implement a firewall-like capability in the OCC.

## E4 SINGLE-SPACECRAFT WITH MULTIPLE GROUND STATIONS

### E4.1 ASSUMPTIONS

This scenario considers a single spacecraft with multiple ground stations. It is assumed that only one ground station transmits to the spacecraft at a time, though transmissions by the spacecraft may be received by both ground stations simultaneously. That is, it is assumed that whatever mechanism is used underneath BP to effect space-to-ground communications supports the equivalent of a multicast destination Data Link layer address. While there may be multiple endpoints on board the spacecraft, and multiple entities on the ground communicating with them (such as a PCC as in the previous scenario), this scenario focuses on the effects of having multiple ground stations.

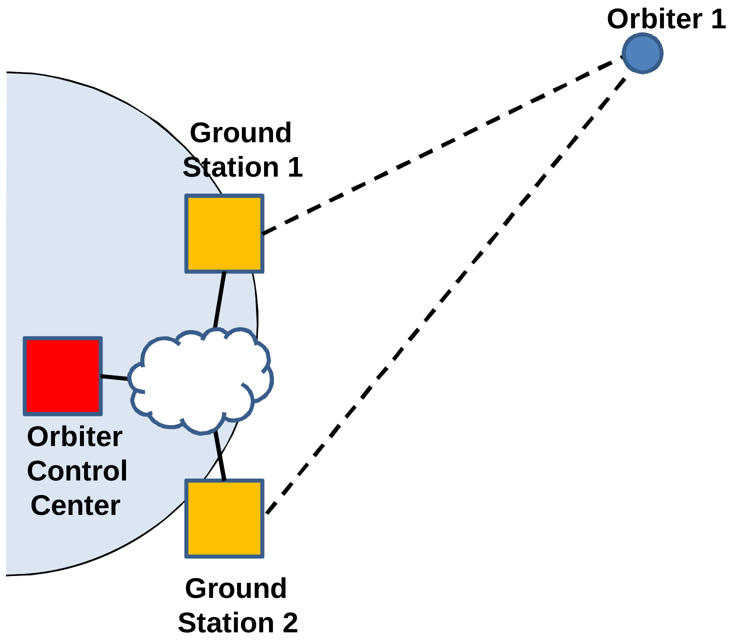


**Figure E-4: Single Spacecraft with Multiple Ground Stations Scenario**

## E4.2 IF THE GROUND STATIONS DO NOT IMPLEMENT BUNDLE ROUTERS

If the ground stations do not implement Bundle routers, then they are transparent from the point of view of the Bundle Protocol. All BP connections are between the OCC and the orbiter. In this case something would have to direct traffic from the OCC into a particular CSTS tunnel to one or the other of the ground stations. The mechanism used to direct traffic into a particular CSTS tunnel is adjunct to DTN; it would need to be configured and managed separately compatibly with DTN contact information so that at the times when DTN transmitted data there would be a path (via one of the ground stations) to the spacecraft.

Mechanisms that could be used to cause OCC-orbiter traffic to be radiated by a particular ground station include:

- some sort of traffic engineering (e.g., IP tunnel) at the convergence layer underneath BP;
- IP multicast to cause traffic to be sent to both ground stations coupled with service management to deactivate one of the ground station transmitters.

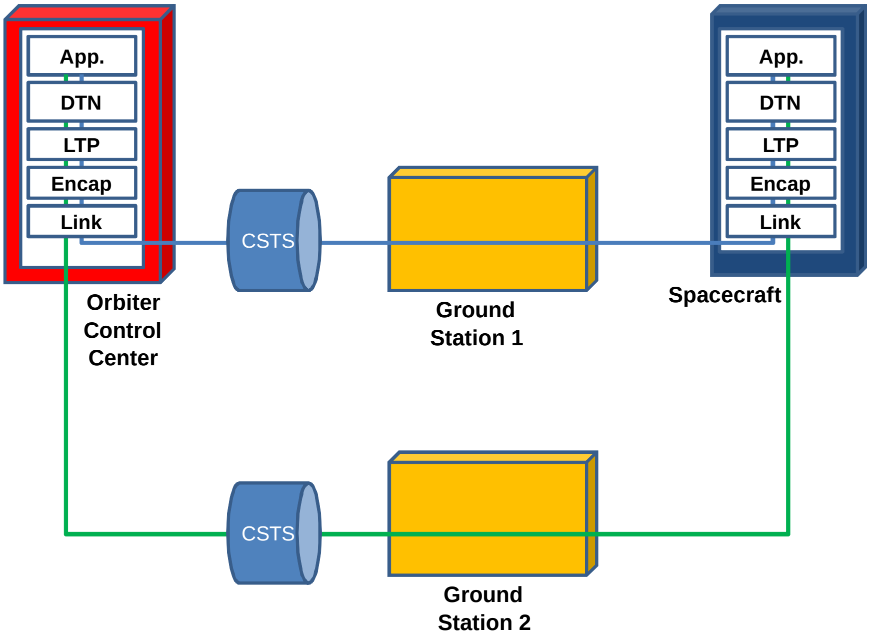


**Figure E-5: Protocol Stack Diagram**

### E4.3 IF THE GROUND STATIONS IMPLEMENT BP ROUTERS

If the ground stations implement DTN routers then the control center can use TCP/IP to communicate with them over the terrestrial network. The terrestrial network connecting the OCC to the ground stations may be a private, closed network that is inaccessible from the Internet.

If the ground stations implement DTN routers, then the connectivity and routing information shown above for OCC-to-orbiter connectivity can be used to control OCC-to-ground-station connectivity. This would provide back-pressure to the OCC DTN implementation, building a queue there when data could not be transmitted from a ground station to the Orbiter.

Alternatively, the contact information between the OCC and the ground stations, and between the ground stations, could be continuous from the point of view of DTN. In this case, the OCC would be able to emit Bundles at any time to one or the other of the ground stations, and it would be up to the ground stations to manage communications with the orbiter. If a ground station were unable to clear its queue of Bundles during a particular contact and the next contact was via the other ground station, the first station might forward its remaining Bundles to the other for transmission during the next contact.

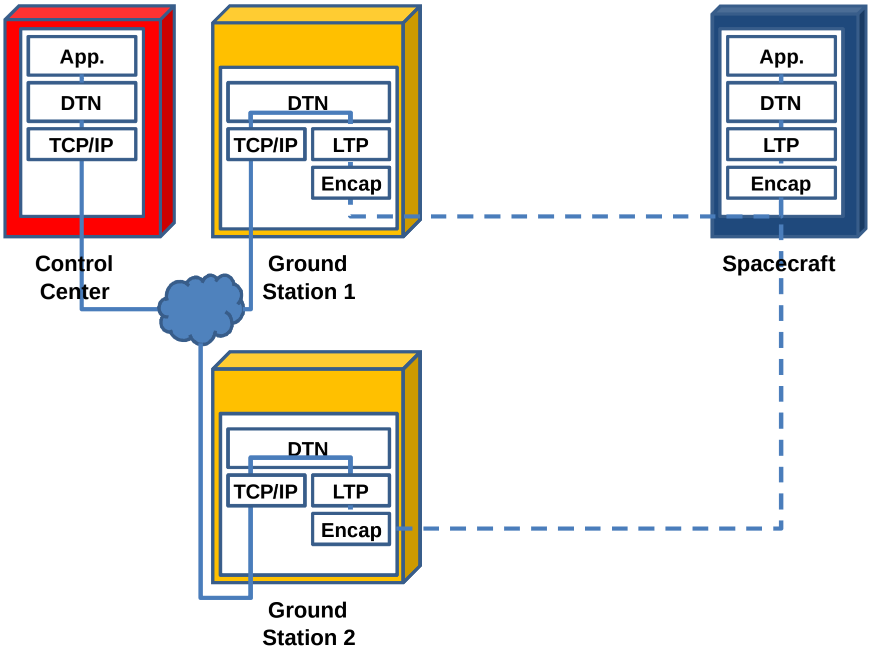


**Figure E-6: If the Ground Stations Implement DTN Routers**

## E4.4 ISSUES

### E4.4.1 Reception of Bundles at Multiple Ground Stations

If both ground stations can receive from the spacecraft simultaneously, multiple copies of the same Bundles will be received by the control center. The Bundle Protocol as specified in RFC5050 does NOT provide duplicate suppression; it is the task of the application, if it requires it, to ignore duplicate Bundles.

### E4.4.2 Choice of Ground Station for Commanding

The choice of ground station for commanding was discussed above separately for the cases where the ground stations do or do not implement Bundle routers.

### E4.4.3 Effects of Unexpected Changes in Connectivity

The results of unexpected changes in connectivity for this example depend on the direction of data flow.

For data transmitted to the Orbiter, unexpected loss of connectivity with a particular ground station will result in unexpected cessation of transmission to the orbiter. The exact mechanism for recovery depends on the capabilities of the BP implementation:

- Reactive fragmentation: if the transmitting BP router (the OCC here) knows how much data from a particular Bundle transmission has been successfully transmitted when contact with the orbiter is lost, it could choose to *reactively fragment* the Bundle. This assumes that the receiver forms a Bundle fragment from the data already received, and the sender forms a fragment from the last known byte received. The fragment at the source is then re-routed, possibly via the other ground station during its next transmission opportunity.

- Link-layer persistence: The Licklider Transmission Protocol can use bidirectional communications to detect link outage and suspend operations when connectivity is lost and resume transmission when connectivity is restored. If LTP is used as the underlying mechanism for OCC-Orbiter communications, then this mechanism can be invoked. In this case the underlying data link is between the OCC and the Orbiter; the ground stations are not involved. Thus LTP transmission could resume as soon as *either* ground station re-established connectivity with the orbiter.

- Abort and retransmit: The OCC may simply abort transmission of the Bundle and retransmit it at the next opportunity.

For data transmitted *by* the orbiter, unexpected changes in connectivity may be less traumatic. If the orbiter loses connectivity with a single ground station but is still able to send to the other, then there are no changes except that the destination stops receiving duplicate copies of Bundles sent by the orbiter. If the orbiter loses connectivity altogether, then the situation is similar to the above case, and the above methods may be used.

## E5 COMMUNICATING WITH A LANDED ELEMENT VIA RELAYS

### E5.1 ASSUMPTIONS

It is assumed that only a single orbiter communicates with the landed element at a time. This could be imposed by the Data Link layer, for example.

For this example only the case where the ground station implements a DTN router is considered.

### E5.2 COMMUNICATIONS WITH A LANDED ELEMENT VIA AN INTERMEDIATE RELAY

#### E5.2.1 General

Here an OCC is in charge of some number of orbiters, and a separate Lander Control Center (LCC) wishes to use the orbiter services to communicate with a remote landed element. This situation generalizes to communicating with any node past the orbiter(s), and to multiple hops in space and/or on the ground.

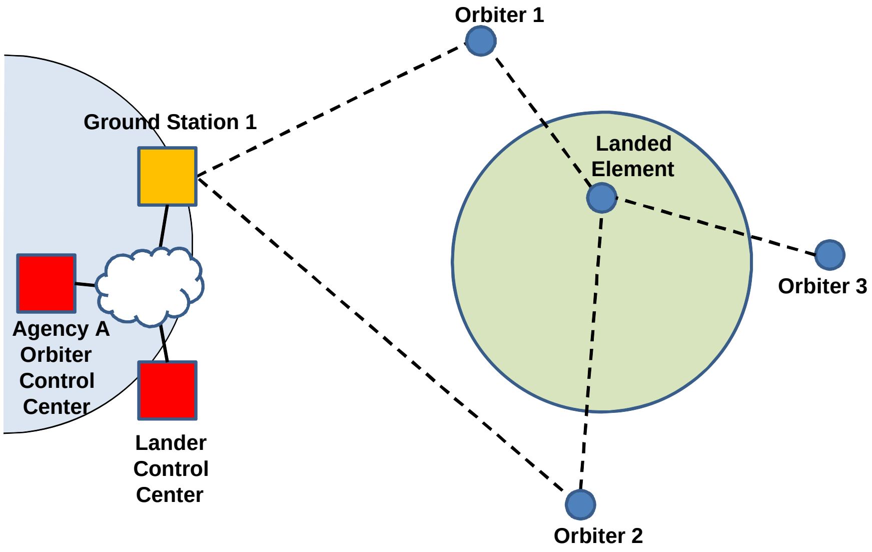


**Figure E-7: Multiple Orbiters Communicating with a Single Landed Element Scenario**

### E5.2.2 Protocol Stack Diagram

Figure E-8 shows a protocol stack diagram for the control center to landed element path. Communications from the LCC may be routed through the OCC as in the above example if policy so dictates.

This example shows LTP used over the Earth-to-space link, and Proximity-1 in reliable mode over the orbiter-to-landed element link. With Proximity-1 in reliable mode there is no need for the reliability mechanisms of LTP.

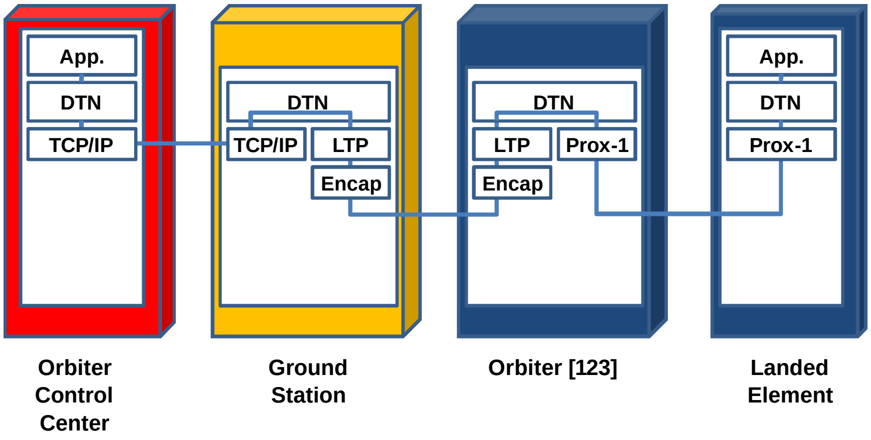


**Figure E-8: Protocol Stack Diagram Including Landed Element**

## E5.3 ISSUES

### E5.3.1 How Does the Lander Control Center Traffic Get to the Lander?

If the LCC is allowed to transmit to the lander without requiring that the commands be checked at the OCC, the LCC would send Bundles directly to the ground station, which would route the Bundles to the next appropriate orbiter. The choice of orbiter is in the purview of the routing/forwarding. This is foreseen as being the typical mode of operation.

If for some reason the orbiter mission operations personnel are concerned that commands addressed to the lander could somehow interfere with the orbiter operations, the lander commands could first be sent to an application in the OCC. This would probably be very atypical, since it would require that the OCC be able to interpret the internals of the lander command traffic (an Application layer protocol).

### E5.3.2 How Does the Lander Data Get to the Lander Control Center?

The Bundle Protocol, as a Network layer protocol, can route lander data to the LCC.

### E5.3.3 How Does the System Function if the Connectivities among Elements Are Intermittent?

There may never be an end-to-end path between the LCC and the landed element.

The Bundle Protocol is resilient against temporary network partitions and cases where end-to-end paths do not exist. Bundles are forwarded hop-by-hop along the paths dictated by the forwarding tables of the intermediate routers, and may be stored at intermediate Bundle routers until a forward path is available.

### E5.3.4 How Does the Ground System Decide to which Orbiter Data Should Be Transmitted?

The Contact Graph Routing mechanisms discussed would allow the ground station to identify the correct orbiter to transmit to. Control over the order in which queued Bundles were transmitted would be under the purview of the BP prioritization mechanisms.

### E5.3.5 How Does the System Decide When Data Should Be Transmitted and When It Should Be Held Waiting for Different Connectivity?

(See above.)

#### E5.3.6 What Happens When the Connectivity between Elements Changes, Either in a Predictable Way, or Not?

If data is transmitted to Orbiter 1 for forwarding to the landed element and that data is for some reason not transmitted during the intended Orbiter 1-lander contact, then the mechanisms described in previous subsections (e.g., reactive fragmentation, link-layer persistence, retransmission) can be invoked. In this case, the remaining data is resident on a particular orbiter and cannot be forwarded to the landed element until the next time the orbiter has contact with the landed element. While it might be possible in the cases of reactive fragmentation and retransmission (though not Data Link layer persistence) to forward the remaining data through the ground station and via a different orbiter to the lander, this would almost surely not be done in practice.

If such ‘data stranding’ events were estimated or found to be common, automated mitigation mechanisms might be devised to attempt to reduce the latencies of ‘stranded’ Bundles. For example, Bundles could be fragmented and, for those fragments that the ground station estimates will be transmitted at the ends of orbiter-to-lander contact periods, duplicate fragments could be proactively forwarded to the orbiter with the next contact. In such cases, multiple fragments might arrive at the lander, but he Bundle Protocol specifies how to reassemble fragments into entire Bundles, and duplicate or overlapping fragments are addressed during that process.

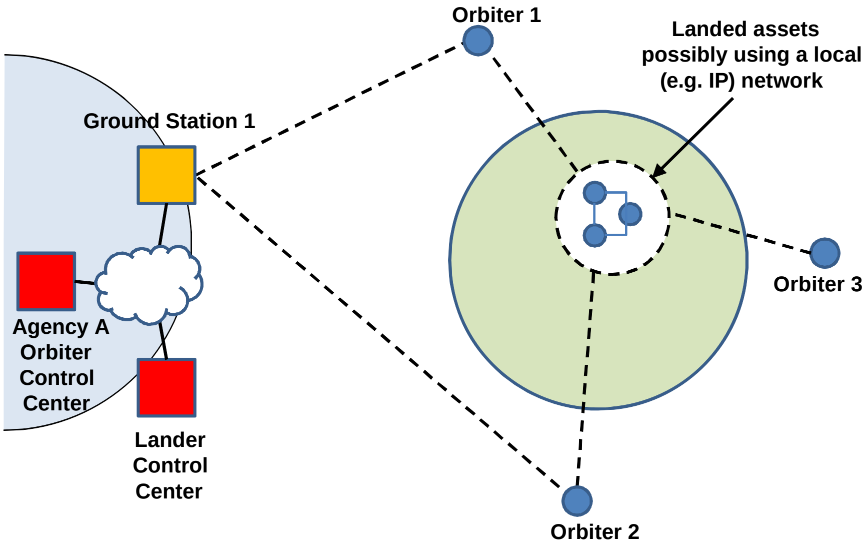
A similar but more complex mechanism would be to use a layer above the Bundle Protocol that implemented forward erasure coding and, for those Bundles with codeblocks at the ends of orbiter-to-lander passes, send a few codeblocks to the next scheduled orbiter.

Both of the above mechanisms assume that the ground station (or the transmitter on the ground if the ground station does not implement a Bundle router) can estimate which Bundles/fragments will be sent at the ‘ends’ of an orbiter-lander contact.

## E6 DTN SUPPORT FOR REMOTE IN-SITU NETWORKS

### E6.1 GENERAL

(See C6.)


Orbiter 1
Landed assets possibly using a local (e.g. IP) network
Ground Station 1
Agency A Orbiter Control Center
Lander Control Center
Orbiter 3
Orbiter 2

## E6.2 ISSUES

### E6.2.1 How Do the Landed Elements Communicate with Earth?

In particular, how does the local networking protocol relate to the protocols used for the surface-to-orbiter and orbiter-to-Earth links?

(See C6.)

# ANNEX F

# TRANSITION TO DTN

This annex describes one possible evolutionary path towards DTN deployment through the use of currently available, published CCSDS Recommended Standards. The aim is to provide a set of services that are functionally equivalent to those provided by DTN, thus allowing development of standard delay/disruption tolerant applications that may be supported by both current and future DTN services. DTN deployment will then be driven by the complex topological demands of future missions and space communities.

Figure F-1 shows the simple present-day scenario of a spacecraft in direct contact with Earth requiring file transfer and manipulation services.

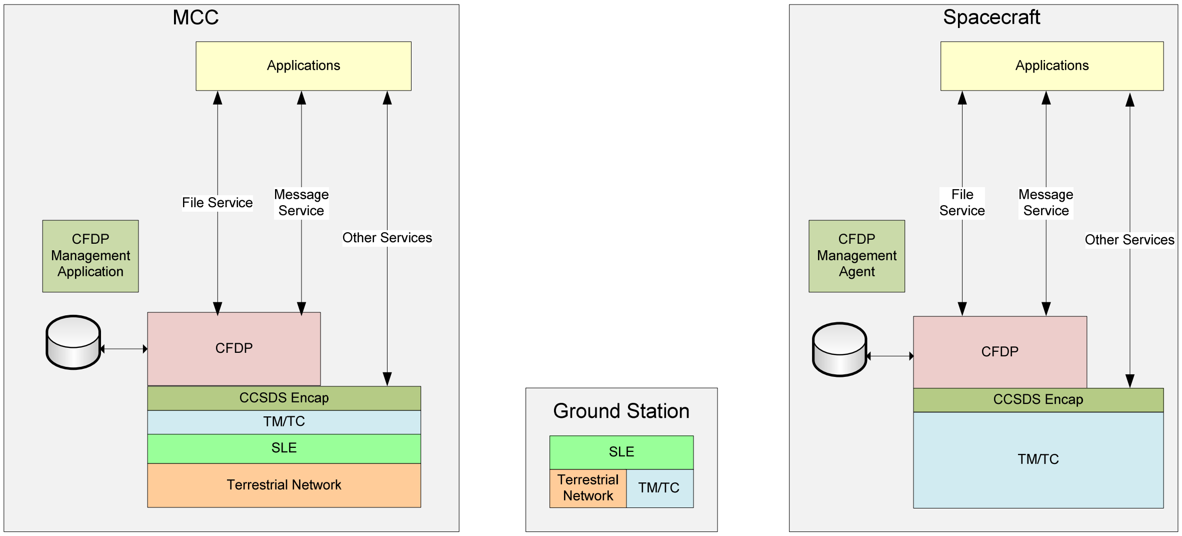


**Figure F-1: Simple Direct Contact Scenario**

In this case, CFDP is used to move files from filestore to filestore under the direction of a CFDP user application. CFDP may also provide a message service to the user applications either associated with a file transfer or not. Other protocols may be transferred using the CCSDS Encapsulation Service. The ground station operates the lower level (framing) of the TM and TC protocols with the higher (packet) layers being at the Mission Control Center (MCC). The ground station services are invoked via Space Link Extension (SLE) services (references [17]-[19]).  A CFDP management application at the MCC may interact with a CFDP management agent at the spacecraft for monitoring and control of the remote CFDP entity. In this case, this is likely to have very little, if any functionality.

Figure F-2 shows the more complex case of operating a lander via an intervening orbiter relay with an interoperating LCC.

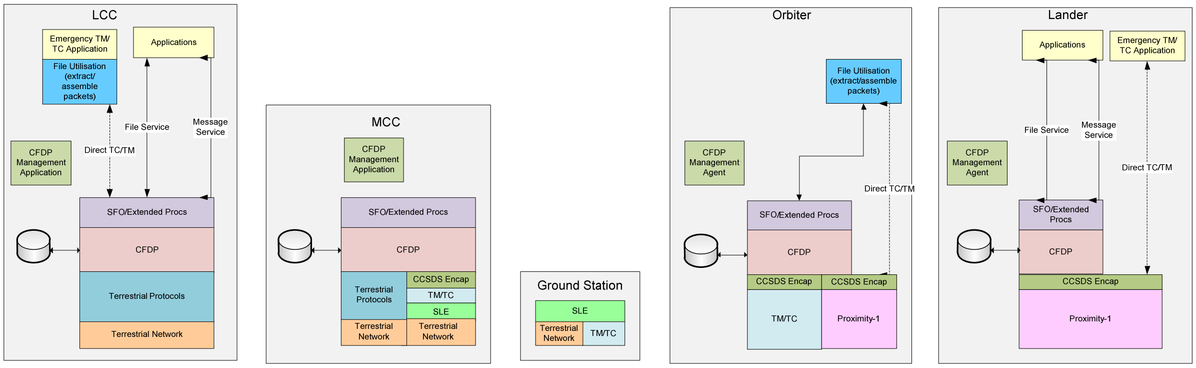


**Figure F-2: CFDP Applied to Orbiting Relay Scenario**

In this case either CFDP SFO or Extended Procedures are used to relay data at the MCC and the orbiter between lander and the LCC. Again, the ground station operates at lower layer protocols.

In addition to the CFDP file and message service, an emergency telecommand/telemetry service is provided via CFDP transactions between LCC and orbiter. At the orbiter, TC frames are extracted from the file and inserted into Proximity-1 as Proximity-1 frames. Likewise, TM frames are extracted from Proximity-1 and packed into a file for CFDP transfer to Earth.

The CFDP management agent at the orbiter allows the MCC to manipulate queues of CFDP PDUs by transaction ID, thus allowing prioritization, preemption, and deletion based on transaction ID and hence by file name. Likewise, the lander management agent allows the LCC to perform similar tasks. A management interaction between LCC and MCC must occur to allow the LCC to request manipulation at the orbiter.

Finally, figure F-3 shows the transition to a BP-based internetworked solution.

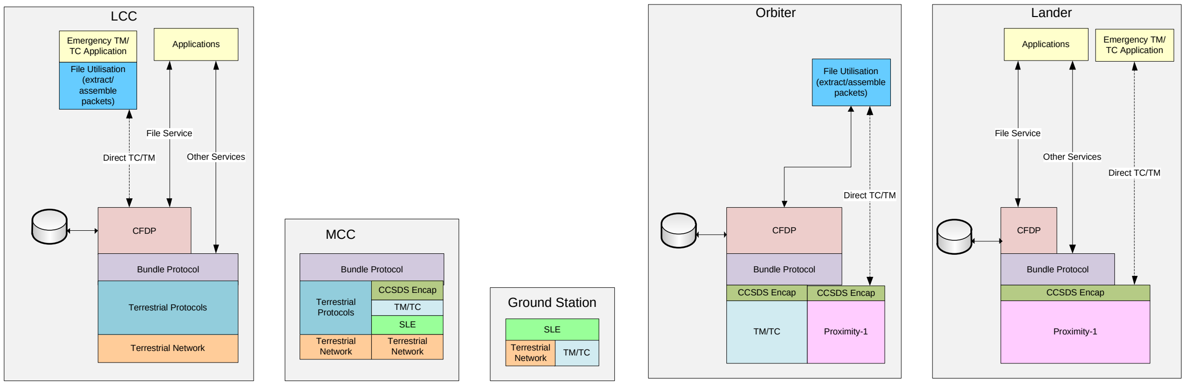


**Figure F-3: BP Solution to Orbiting Relay Scenario**

In this configuration, BP is used to provide storing and forwarding at the orbiter. CFDP is used at the end systems to provide file services and at the orbiter to support the emergency

TM/TC capability.  BP provides a general purpose store and forward service which can support other end-to-end services (e.g., packet, message).

The figure shows TM/TC in the long-haul link. These could be supplemented by LTP in the time frame of this configuration.

The above description shows it is possible to implement a smooth transition to BP deployment by providing standard messaging and file transfer services which can be implemented using both current and DTN protocols. The orbiter and lander applications for provision of emergency commanding and telemetry can also be implemented independently of the underlying network service. As more complex mission topologies than those currently being implemented emerge, it will thus be possible to adopt DTN protocols with minimum disruption to Application layer development.