

Applying the New CCSDS File Delivery Protocol in a Mission Environment: A Case Study of the JPL TT&C Ground System Adaptation for the Deep Impact Mission

Michael Levesque⁽¹⁾, Felicia Sanders⁽¹⁾

*(1) Jet Propulsion Laboratory
M/S 230-305
4800 Oak Grove Drive
Pasadena, CA 91109
USA*

*Email: Michael.E.Levesque@jpl.nasa.gov
Felicia.Sanders@jpl.nasa.gov*

ABSTRACT

The emerging Consultative Committee for Space Data Systems (CCSDS) File Delivery Protocol (CFDP) standard will re-shape ground support architectures by enabling applications to communicate over the space link using reliable-symmetric transport services. Traditional ground support architectures for telemetry and commanding will need to operate as link-level devices rather than in the application domain as typically done when applications communicate using CCSDS transfer frames or packets. The Jet Propulsion Laboratory (JPL) is implementing the CFDP standard to support future missions including Deep Impact. The new architecture is based on layering the CFDP applications on top of the new CCSDS Space Link Extension Services for data transport from the mission control centers to the ground stations. This paper uses the Deep Impact mission as a case study and discusses the architecture, requirements, and design for integrating the CFDP into the JPL Deep Space Mission System (DSMS) ground support services.

INTRODUCTION

This section provides an introduction to the CFDP, the Deep Impact Mission that is likely to be the first mission to fly the CFDP and the JPL Deep Space Mission System (DSMS) Tracking, Telemetry & Command (TT&C) system that will be providing the CFDP ground system services to the Deep Impact Mission.

The CCSDS File Delivery Protocol

The CCSDS File Delivery Protocol (CFDP) is a specification describing an open, international standard file delivery protocol. CFDP provides the mechanism to transfer files to and from a remote entity using a local entity. The content of the files is independent of the protocol used for the transfer them. Files can be transferred reliably, where it is guaranteed that all data will be delivered if transferred without protocol error, or unreliably, where a best effort transfer is performed. Files can be transmitted with a unidirectional link, a half duplex link, or a full duplex link, with near-Earth and deep space delays. File transfers are controlled through the user application and can be triggered automatically or manually. Fig. 1 is a sequence diagram illustrating the file transfer protocol [1].

The CFDP Flight-Ground baseline is a JPL software implementation of the CCSDS specification that when integrated into a complete flight software system or ground software system, provides point to point reliable and unreliable file delivery between systems [2]. This CFDP implementation provides a fault tolerant communication mechanism that operates efficiently over the highly constrained physical links encountered in deep space communication. It is designed and implemented to have full error recovery from system resets. The CFDP architecture and implementation allows it to be integrated and executed on a variety of hardware platforms including the highly constrained platforms often required for deep space flight systems.

The CFDP (Flight-Ground baseline) implementation consists of three functional assemblies: (1) the Protocol Daemon, (2) the Adaptation Layer and (3) the Test Harness. The Protocol Daemon assembly is a process that is responsible for implementing the CFDP protocol functional requirements. It processes Protocol Data Units (PDUs) in real-time and

- (1) extracts PDU inputs from the remote CFDP entity,
- (2) formats PDU outputs destined for the remote CFDP entity,
- (3) implements the CFDP standard rules and other requirements within the specification,
- (4) utilizes the Adaptation Layer software for platform independent system resources,

- (5) communicates with the local CFDP user software using the Application
- (6) communicates with the local User Transport (data link) software.

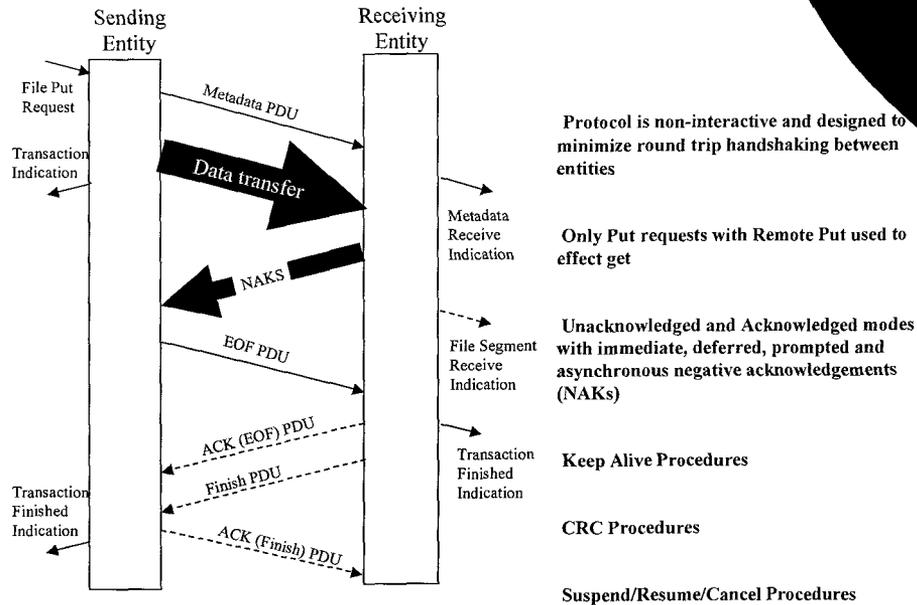


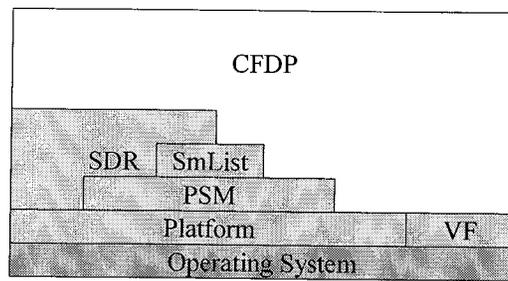
Figure 1. Protocol Sequence Diagram
Report Procedures

The Adaptation Layer consists of the software and its software libraries that enable the functional requirements to be deployed independent of the operational environment. It includes communications services, data services (pipes, queues, lists), file services (file manipulation and control, file storage management), controls (semaphore), etc.

The CFDP architecture including the Protocol Daemon and Adaptation Layer components are shown in Fig. 2.

The Test Harness provides a 'CFDP User Template' for testing the CFDP baseline. The Test Harness will implement basic functions to communicate with the Protocol Daemon only, using the Adaptation Layer software. The purpose of the Test Harness software is to send requests and accept responses from the Protocol Daemon.

In a deployed application, the Test Harness assembly would be replaced by a user application that interfaces with the external operational environment. Its functions are to (1) access the underlying file system, (2) queue and manage multiple, simultaneous CFDP standard files (inbound, outbound, and Management Information Blocks (MIBs)), (3) monitor the radio frequency (RF) activity, (4) collect accountability statistics and report.



CFDP	Protocol Daemon = fdpd, fdpi, fdpo, fdpq
SDR	Simple Data Recorder = persistent object database in shared memory, using PSM and SMList
SmList	linked lists in shared memory using PSM
PSM	Personal Space Management = memory management within a pre-allocated memory partition
Platform	access to operating system such as shared memory, system time, pipes, queues, semaphores
VF	Virtual Filestore = isolates CFDP from file I/O
Operating System	thread/spawn/destroy, file system, time, inter-process communications

Figure 2. CFDP Architecture

The CFDP baseline data flow, which will be integrated into the DSMS TTC&DM system services, including the Protocol Daemon, Adaptation Layer services, User Application and UT Layer interfaces, are shown in Fig. 3.

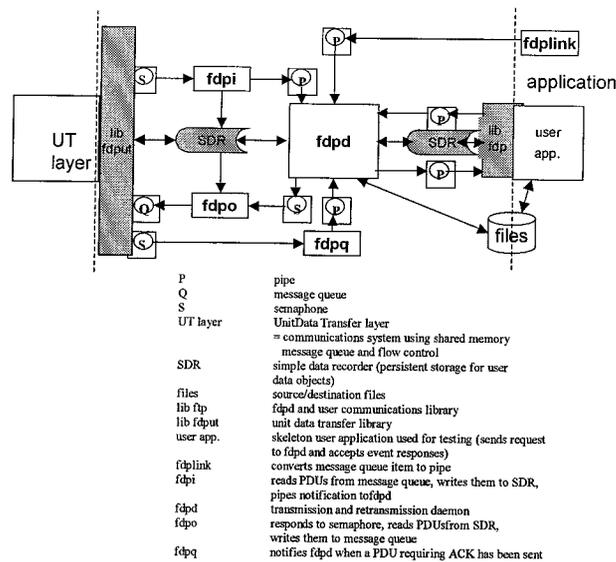


Figure 3. CFDP Baseline Data Flow

The Deep Impact Mission

Scheduled for July 2005, Deep Impact's spacecraft will arrive at comet Tempel 1 where an impactor will be used to create a large crater. This attempt to peer beneath the surface of a comet to its freshly exposed material will be captured in images from cameras on both the impactor and flyby spacecraft which will be sent back to earth in near real-time.

The mission as adopted the CFDP for the majority of its data transport including relaying information from the impactor through the flyby spacecraft. The mission hopes to simplify on-board data handling, ground processing to reconstruct science products and automate reliable uplink file transfers. While still work in progress, flight/ground interfaces and operational scenarios are being developed to provide these capabilities through the CFDP.

Uplink Scenario

The Deep Impact Mission plans to use traditional command services that are non-CFDP uplinks with a unique command virtual channel for Hardware CRC commands and another command virtual channel for software immediate commands. There is no Deep Impact scenario to interleave a CFDP uplink with commands. There is also no Deep Impact scenario to suspend and resume CFDP uplinks for the purpose of interleaving commands. It will be necessary to queue a mixture of different CFDP uplinks and command uplinks in sequence. All uplinks will occur within a single station uplink activity, so the mission has no need to split a uplinks across multiple station tracking sessions. File reception status will be reported in telemetry but at a minimum files not received correctly through CFDP will never reach their final destination user application on-board the spacecraft. Files to be uplinked are stored on the Data Management System on the ground. The ground user application will be used to select and queue approved/ scheduled files for uplinking either as command or through CFDP. For commands radiation success or failure status is provided while for CFDP completed on-board file transfer success/ failure status is provided back through the user application.

Downlink Scenario

The Deep Impact Mission plans to use CFDP for the bulk of its data transfers from both the flyby and impactor spacecraft. Merged with the CFDP data will be traditional real-time CCSDS telemetry frames containing packets for each spacecraft on separate virtual channels. CFDP PDUs may also contain CCSDS packets as stored into files on-board the flyby spacecraft. Storing packets into files and transferring these files eliminates the need for individual packet queues, packet queue management and packet queue transport priority tables as is found in traditional deep space flight software. Under nominal mission scenarios the data access latency built up with transfer packet data as files is not prohibitive to the mission plan or spacecraft health and safety. During non-nominal periods this latency can pose problems and packet data contained in CFDP PDUs should be made available to application software as they are received. Engineering telemetry packets in CFDP PDU's should be processed and displayed as they are received. This is important for analysis and recovery of safe- mode conditions at low bit rates. Finally, file reception status must be made available to the OPS team, so that incomplete files can be scheduled for re- transmission, and partially received files can be analyzed. This is important in the "live-for-the-moment" philosophy of the flyby mission, since the spacecraft may not be able to re- transmit science images.

The TT&C Ground Data System

JPL supports unmanned deep space missions through the DSMS Advanced Multi-Mission Operations System (AMMOS). The AMMOS supports all JPL deep space mission flight operations. There are several service domains within the AMMOS, including Telecommunication Services, Network Infrastructure Services, Mission Sequencing and Analysis Services, Navigation Services and Science Information Services. The Telecommunication Services consist of Tracking, Telemetry, Command and Data Management (TTC&DM) mission operations services. Components of the AMMOS TTC&DM include flight and support equipment data acquisition, telemetry processing including frame synchronization, decoding, and channelization, telemetry data cataloging, archiving, retrieval and distribution, engineering data processing for converting data numbers to engineering units, alarming and displaying engineering data, and various other tools for analyzing and displaying telemetry data as shown in Figure 4. It is these services that will be integrated with the CFDP file delivery services as described in the following sections.

the assistance of Data System Operations personnel for protocol daemon monitoring and fault diagnosis. During uplink file transfer transactions, the TTC&DM Command Subsystem and operations personnel will be responsible for identifying files to be transferred and establishing and maintaining the command link with its DSN counterpart. Downlink file transfer will occur automatically and be recorded in the Data Management System.

The following is a summary of the key end-to-end operational scenario associated with the new CFDP file delivery service:

- 1) Each CFDP baseline entity will operate as a process daemon and contain all of the functionality required to perform the CFDP protocol in the point-to-point service mode.
- 2) Each CFDP entity will allow for the connection of a single user application for the implementation called the session manager.
- 3) Each CFDP baseline entity will have a process started to receive telemetry packets containing CFDP PDUs, extracting those PDUs and outputting them to the CFDP entity.
- 4) The User Application will interface with the Data Management System for cataloging received files from the CFDP entity.
- 5) The User Application will provide a control interface to the AMMOS Command Radiation Service for controlling the uplink of files.
- 6) The User Application will interface with the Data Management System for sending files to the CFDP entity.
- 7) The AMMOS Command will start a process to read CFDP uplink PDUs, perform the Telecommand wrapping function and insert them into the Space Link Extension CLTU service.

ISSUES AND RESOLUTIONS

As with most CCSDS standards the specification allows for interoperability within the protocol layer but does not specify how applications within a specific mission use the communications protocol. Such is the same with the CFDP standard. The issues and resolutions summarized in this section are provided from experiences with integrating the CFDP into the Deep Impact Mission Operations System.

Asymmetric Control between Uplink and Downlink

Most Deep Space Mission Operations Systems (MOS) require operations control over the uplink path while downlinks are typically automated through sequence generation. This asymmetry is amplified when integrating a symmetric transport protocol such as CFDP because only one side of the communication must be controlled regardless of the type of traffic, eg. Uplink, Uplink Re-transmission, Downlink Acknowledgement. Further exasperating the problem is traditional store and forward deep space uplink communications that must co-habitat in the uplink channel with automatic file delivery including automatic re-transmissions and acknowledgements.

To alleviate the first issue, Deep Impact end-to-end mission operations design has chosen to uplink files reliably but downlink files unreliably. By using unreliable downlinks, no uplink command traffic is generated or necessary for the protocol to operate. Thus uplink traffic or commands that are driven by the protocol to satisfy downlink transactions do not have to share the uplink bandwidth nor require MOS control. Furthermore downlink file transfers through CFDP are performed automatically in the ground system while the flight system uses sequences and traditional commanding to determine the downlink transactions that are initiated on-board. Thus further simplifies the MOS control by not requiring remote file operations including remote puts to be initiated from the ground system which for deep space missions with long one-way light times would be further prohibitive.

As outlined above in the GDS introduction the AMMOS Command Subsystem provides a store and forward command service using a Graphical User Interface (GUI) and communication with the DSN over SLE CLTU. The Command GUI allows a mission controller to select, control radiation and determine radiation status of near real-time commands and stored commands in the form of sequences. This poses the second issue and in order to integrated CFDP services the Command GUI will be augmented to include store and forward transport of files through the CFDP and subsequently over the SLE CLTU. The Command GUI will serve as the CFDP user application communicating through a session manager for uplink file transactions. File transfers through CFDP will maintain state within both the CFDP daemon and the GUI or session manager from DSN pass to pass. In this configuration different uplink transfers, near real-time command and sequences, are controlled in first in first out (FIFO) queues and CFDP files transfers are maintained in separate transaction queues that are initially uplinked in FIFO with retransmissions and other PDUs for open transactions sent as first priority. Thus the MOS will maintain traditional commanding capabilities while augmenting these with reliable file uplink. Transmission latency, through the SLE CLTU, is controlled by providing transfer windows that open immediately when received by the provider. Transfer latencies between traditional

commands and CFDP file transfers is kept to a minimum by using two output queues with control provided to the mission controller.

Partial File Handling

By specification CFDP does not return or create files if there is a protocol error due to file checksum or file size failure. The CFDP does provide for PDU data segment indications to the user application for returning file data as each segment arrives. Since downlink transactions are performed in un-reliable mode the ground system must provide for processing of partial files or files that result in a CFDP protocol error since deep space downlink margins are often small and errors often occur. This partial data is returned by using the file data returned in the PDU data segment indications and creating a file with a type, length, offset, time, data value (TLVO) structure for each returned data segment. Data segments are appended to the file in arrival order regardless of duplication or gaps in the data. These TLVO structures can be parsed by multi-mission software to create a best efforts file with fill added for gaps and duplicate data merged. Alternatively these TLVO files can serve as a transaction log for files that are transferred without error and are created by CFDP.

File Identification

Using unreliable file transfers not only makes handling partial files important but also requires handling transactions that don't have meta-data associated with them. The meta-data is critical in that it carries the destination file name but in unreliable transfers this data can be lost along with other file data. Fortunately each PDU contains a unique transaction sequence number which will be used to specify destination files names in lieu of receiving the meta-data PDU. Naming files is one issue but identifying what the file contains is yet another. It has been proposed to overload the sequence number, which by specification is not restricted to be other than a unique 16-bit value, with a method for identifying the file among its peers. It is not hard to conceive of such an identifier but this would complicate the mission and ground software interface. Therefore given the Deep Impact mission operations plan to sequentially transfer files the file creation time will be used to correlate an unknown file with the actual file name and type sent by sequence control on-board software.

Near Real-Time Downlink Data Access

Storing data in files and transferring them through CFDP simplifies on-board data handling and ground file reconstruction. But dealing with data as file products also adds some latency to data availability since you nominally wait until the file transaction is completed before accessing the data in the file. For higher data rates and small files this latency is not a problem. Nominally the Deep Impact mission will be using two mega-byte files and transfer data at 80 kbps resulting in a latency of up to 200 seconds or over three minutes. Nominally this is not a problem given the latency of pre-storing the data on-board and given one-way light times. But in the case of emergency mode operations when the spacecraft may be in safe mode and transmitting at data rates of 40 bps the latencies incurred in transferring files becomes dominant. Under these conditions, small files help reduce latency yet increase on-board processing and access to file data segments as they arrives further reduces latency at the expense of increasing ground complexity. The Deep Impact mission has chosen to employ both methods, using small files on on-board during safe mode operations and ground processing of file data segments in real-time.

In order to process the file data segments in real-time the ground user application must have access to the data and further be able to determine how to process it. As discussed earlier the CFDP data segment indication will return each file data segment as it arrives. Using this capability and a segmentation control capability within CFDP to process data records on known boundaries, in this case CCSDS Telemetry packets, the flight CFDP entity will place the CCSDS packets into PDUs on integral number boundaries. The ground CFDP entity will use the segmentation control flag, contained in every PDU, to know to extract the CCSDS Telemetry packets from the PDU data segments and further process them for mission operations in near real-time as required by Application Process Identifier (APID).

Relay Data Handling

The CFDP specifies and implements point to point file delivery. For relay operations, this requires at least two file transfers, one between the source entity and the relay entity and a second between the relay entity and the destination entity. For the Deep Impact mission the three entities are the DSMS, the flyby spacecraft acting as the relay and the impactor spacecraft. Unfortunately most flight telecommunications systems do not have processing capabilities to handle CCSDS telemetry and telecommand link layer protocols symmetrically that is both capabilities as both uplink and downlink in the same telecommunication system as is the need when that system acts as a relay. Such is the case for the Deep Impact spacecraft. Thus impactor telecommand data must be created on the ground and stored in files, in

this case with duplicate protocol data units, transferred to the flyby spacecraft and stored as impactor telecommand frames. These are subsequently copied and sent through the flyby telecommunications system to arrive as spacecraft commands at the impactor. The downlink side will require the flyby spacecraft to store impactor telemetry frames containing CFDP PDU's for subsequent transfer through the flyby entity CFDP. The ground will first process the data through a flyby CFDP destination entity and then an impact CFDP destination entity to create the original impactor file. CCSDS Telemetry Synchronization of the impactor data will be critical for extracting the CFDP PDU's consistently.

FUTURE WORK

To date we have addressed most of the high-level mission operations issues with using the CFDP for the Deep Impact Mission. As development, test and operational deployment progress we will find additional issues that need to be resolved most notably in the area of file product design and time correlation.

Future missions will want to benefit from that Deep Impact lessons learned while increasing the operational capabilities available under CFDP. These operational capabilities will pose new mission operation system challenges in the areas of automated symmetric control of file transfers, bandwidth sharing and file data access standards from ground data management systems possibly through new Space Link Extension protocols.

CONCLUSION

In this paper we have addressed many of the Mission Operations System issues with deploying the CFDP. These issues result from traditional forms of mission operations in dealing with the non-traditional modes of operations presented by CFDP. We've shown that through careful integration the MOS can maintain traditional operations and be augmented with CFDP capabilities. Without such integration the CFDP would not have its first targeted mission, the Deep Impact Project. Finally though careful adoption and integration the CFDP capabilities and when fully utilized stand to help reduce mission operation complexity.

REFERENCES

- [1] CCSDS 727.0-R-5:: CCSDS File Delivery Protocol (CFDP). Red Book. Issue 5. August 2001.
- [2] JPL Intellectual Property NTR No. 30362, CFDP Flight-Ground Baseline, Michael Levesque, Aug 2001.
- [3] CCSDS 201.0-B-3: Telecommand Part 1 -- Channel Service. Blue Book. Issue 3. June 2000.
- [4] CCSDS 202.0-B-2: Telecommand Part 2 -- Data Routing Service. Blue Book. Issue 2. November 1992.
- [5] Telecommand Part 3 -- Data Management Service. Blue Book. Issue 1. January 1987.
- [6] CCSDS 912.1-R-2: Space Link Extension - Forward CLTU Service Specification. Red Book. Issue 2. May 2000.
- [7] CCSDS 102.0-B-5: Packet Telemetry. Blue Book. Issue 5. November 2000.

ACKNOWLEDGEMENTS

The work described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

We would like to acknowledge those who have contributed to this work: Scott Burleigh, Alan Schlutzmeyer, Jeane Stipanuk, Sandy Krasner and Son Ho.