Standardizing an End-to-End Accounting Service" – DRAFT 58040

Edward Greenberg[*] and Greg Kazz[†]
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, 91109

Currently there are no space system standards available for space agencies to accomplish end-to-end accounting. Such a standard does not exist for spacecraft operations nor for tracing the relationship between the mission planning activities, the command sequences designed to perform those activities, the commands formulated to initiate those activities and the mission data and specifically the mission data products created by those activities. In order for space agencies to cross-support one another for data accountability/data tracing and for inter agency spacecraft to interoperate with each other, an international CCSDS standard for end-to-end data accountability/tracing needs to be developed. We will first describe the end-to-end accounting service model and functionality that supports the service. This model will describe how science plans that are ultimately transformed into commands can be associated with the telemetry products generated as a result of their execution. Moreover, the interaction between end-to-end accounting and service management will be explored. Finally, we will show how the standard end-to-end accounting service can be applied to a real life flight project i.e., the Mars Reconnaissance Orbiter project.

Nomenclature

CCSDS	=	Consultative Committee for Space Data Systems
CCR	=	Command Change Request (a proposed Activity)
CDS	=	Command Delivery Service
CFDP	=	CCSDS File Delivery Protocol
CLTU	=	Command Link Transmission Unit
CRS	=	Command delivery Service
DISA	=	DSMS Information System Architecture
DSMS	=	Deep Space Mission System
GUI	=	Graphic User Interface
IFL	=	Interactive File Load
NIFL	=	Non-Interactive File Load
NIPC	=	Non-Interactive Payload Command
PDU	=	Protocol Data Unit
SCR	=	Sequence Change Request
SMS	=	Service Management System
SPS	=	Service Preparation System
XML	=	Extensible Markup Language

[*] Principal, Software and Information System Engineering, 4800 Oak Grove Dr. Pasadena, CA 91109 MS 301-265.

[†] Senior Engineer, Software and Information System Engineering, 4800 Oak Grove Dr. Pasadena, CA 91109 MS 301-170U.

I. Introduction

This paper consists of the following sections:

1) Framework Objectives
2) End-to-end data accountability model which includes:
   a) End-to-End functional overview
   b) Information management approach
   c) End-to-end Information relationship model
3) Sequence Planning
4) Sequence Development
5) Command Generation
6) Command Delivery
7) Spacecraft Execution and Reporting
8) Telemetry Services
9) Accounting Services

A. Framework Objectives

1. This framework is a conceptual model for the end to end process.
2. This model includes a proposed Consolidated Command Delivery Service that provides for the automated handling of retransmission activities associated with the CCSDS File Delivery Protocol (CFDP) for both uplink and downlink support.
3. This framework provides for the acquisition of data from each of the elements of the End-2-End mission operational phases for use in monitoring performance of the missions planning and operations processes, the supporting multi-mission services, the spacecraft sequencing activities and explicitly for tracing mission products from planning to final delivery.
4. This framework maintains the relationships between the planning and production processes to enable operational control of the sequence development process and to provides the means for adding planning rationale, creation metadata and collective relationships to the informational products traced with this process.
5. This framework describes at a relatively high level the information that is to be captured at selected points of the End-2-End processes based on the capabilities projected for the in development DISA Information Bus and Accounting Service. The implementation provides for piecewise development of capabilities, enabling Service agents to be added over time to extend the capabilities without interfering with existing capabilities.
6. The GUIs provide visibility and support to various operational activities that will all be accessing the same data and thus their implementation could have common software foundation.
7. The concept integrates the control steps associated with the work flow approval processes for mission sequence development through the uplink process
8. The Consolidated Command Delivery Service is designed for automated production and data capture of the uplink with the creation of an automated summary of uplink activities within a track. This, in addition to the inclusions of operational controls for CFDP utilization could significantly change the role of the flight controller.
9. The framework provide End-2End Accounting for:
   1. Requests/Proposals
   2. Sequence Development
   3. Command Generation
   4. Command Delivery
   5. Spacecraft Activity monitoring
   6. Creation and delivery of Observation Data Products (Telemetry Files using CFDP)
   7. Annotation of Observational Data Products with metadata acquired via the Request/Proposal process

B. End-to-End Functional Overview

Figure 1 below provides a comprehensive overview of the end-to-end information system uplink and downlink functions to which the accounting system interfaces. End-to-End accountability is involved in tracing activity requests (the earliest form of a science or engineering goal conceived on the ground what the spacecraft eventually executes) generated by Sequence Planning, defined and tested as sequences by Sequence Development, controlled by Mission Command Management, delivered to the spacecraft by Command Delivery, executed by Spacecraft Execution and reported and accounted in telemetry by the Telemetry Services.

More later…

Figure 1. End-to-End Uplink/Downlink Functional Overview

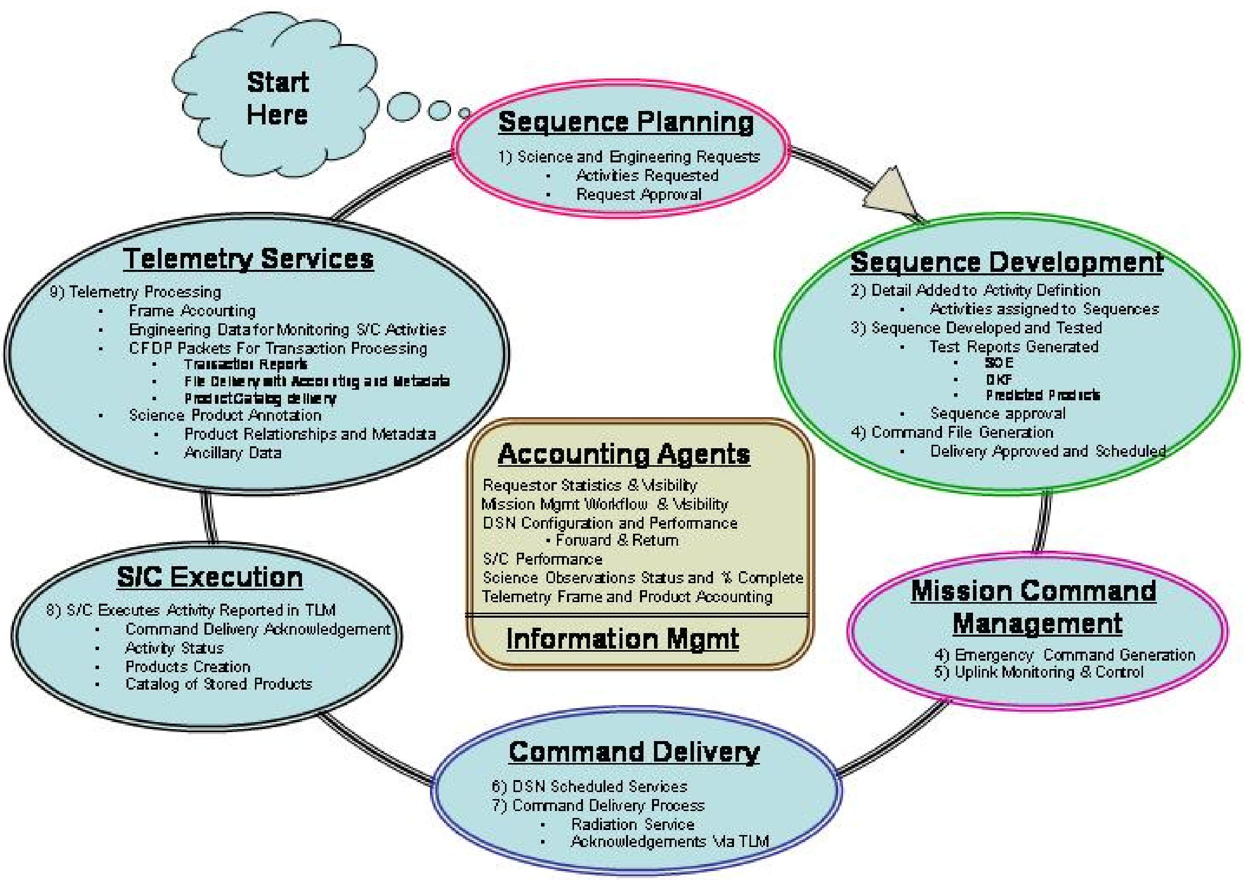
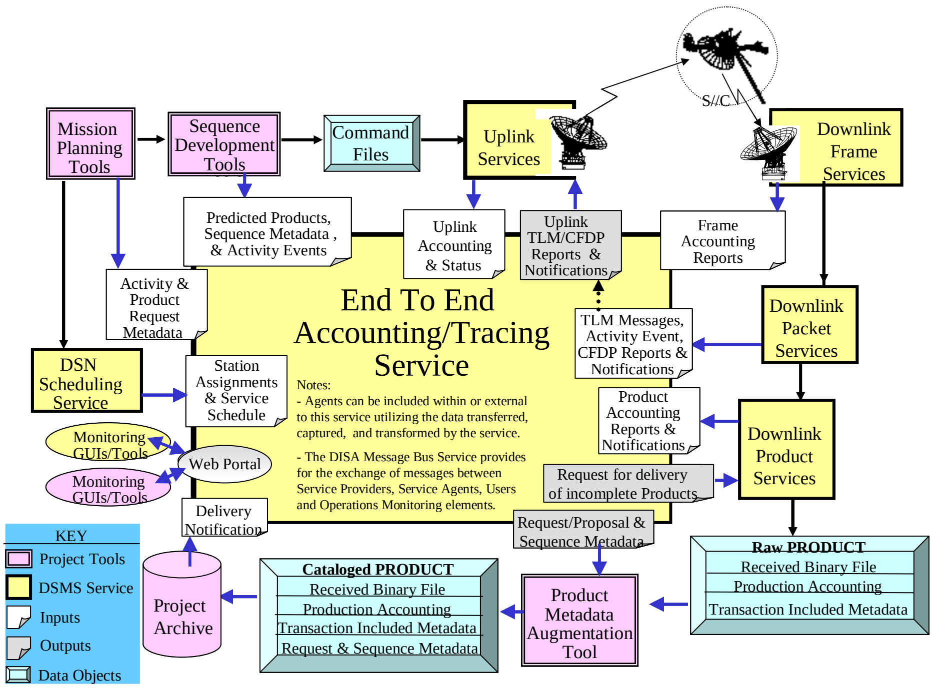


Figure 2: End-to-End Accounting Context Diagram



C. Information Management Approach

1. External Access is provided via a Web Portal which provides Authentication and Authorization
   a. Loading of Files to the File Store
   b. Delivery of control and configuration messages
   c. Acquisition of Files, Records and Event messages are available by user subscription or query

2. XML Command Files are stored as “Read-Only” after receipt from one of the following:
   a. Sequence Planning
      1. Requests (containing one or more Activities)
   b. Sequence Development
      1. Sequences (Implementation of one or more Activities)
      2. Sequence of Events (includes DSN Key Words)
      3. Predicted Products
   c. Command Generation
      1. Command File (Implementation of one or more Sequences)
   d. Service Preparation System
      1. Service Pass Configurations
      2. Light Time Tables
      3. S/C Clock vs UTC (not sure if needed)

3. Accounting Agents read the XML Files and create the following types of DBMS records
   a. Request
   b. Activity
   c. Sequence
   d. Command
   e. Uplink Service Instances
   f. Downlink Service Instances
   g. Data Product

4. Uplink Accounting Agent receives Events, Reports, Logs and Status from Information Bus and updates DBMS records as required
5. Uplink Accounting Agent produces an archival product for each uplink pass
6. Downlink Accounting Agent provides accounting for link operations and Telemetry Services

D. End-to-End Information Relationship Model

Figure 3: End-to-End Information System Relationship Model

II. Sequence Planning

The functionality contained within the Sequence Planning oval in Figure 1 is described below:

1. The data provided in the Request File Object will be utilized to provide the relationship between the Requestor and the Sequence and Data Products resulting from the Request File Object. Thus a simple tool could provide the Requestor with knowledge about the status of the Request, the resultant Sequence and the products produced by the sequence.

2. This process may eventually become more multi-mission but this model considers the Science planning tools as mission specific requiring only information about a request and the activities covered by the Request. The use of an XML format for the Request File Object allows the project/Requestor to include other unique data that their processes require.
3. The data desired from the Request File Object includes:
    a. Request ID

    b. Requestor
    c. Activities (observations) desired
        i. Each Activity is provided an Activity ID
        ii. Relationship between Activities are defined (observation product relationships)
        iii. Time window for each activity or the collection of activities
        iv. Metadata for Annotation of the created products (including name when appropriate)

4. Agents can easily be added to the system to track Requests, Activities and Products for Requestor and Project Management, providing visibility into the status of the Sequence Development Process and/or Product Status and statistics.

5. Work Flow Controls and Visibility can be provided by the Accounting Agent if desired by the project.

III. Sequence Development

Figure 4: Overview of the Sequence Development Process

The following is a description of the Sequence Development Process shown in Figure 3 above:

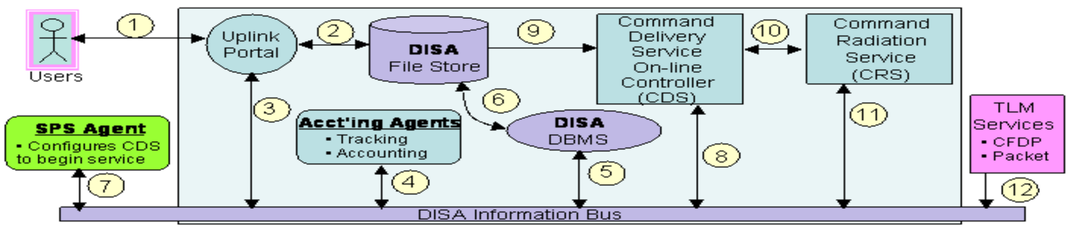
1. CCR or SCR submitted for Request XYZ (Reference name for Request)
    1. CCR Object loaded to File Store
    2. Uplink Request Record Created (XYZ)
        1. Selected data copied to Request Record
        2. Activity Records Created for Request referenced to Request Record
2. Mission Mgr meeting –select Request Records which have not been approved .. Get description of requested activities from CCR Object
    1. Approval updates Uplink Request Record…Approval time date, etc added (also to Activities Records)
3. Sequence Development Team Selects Activity Records that are approved but not yet assigned for development
    1. Assigns developer(s) and provides target delivery dates for the to be produced Command Files, that are added to the Activity Record(s)
4. Activity Designer creates a detail description of activity for translation into command sequence
    1. Designer submits detail description object to file store– submittal date added to Activity Record
5. Activity Sequence Designer creates activity sequence for execution on S/C
    1. Activity Sequence Object is submitted to file store…
        1. Activity Sequence Object is simulated/tested/evaluated
            1. Test results generate Activity Ancillary Objects (include SOE, Predicted Products, pertinent parameter info)
6. Design Approval Meeting is held…Access various objects to determine Activity sequence performance
    1. Acquire Request, Activity Detail Description, Activity Sequence Specification and Test result objects from store
    2. Acquire mission timeline and DSN Pass Instances then evaluate and note Delivery Pass ID and Priority within pass
    3. ***--------------------- Iterate process D-F until approval….each iteration is versioned -----------------***
7. A Command object that implements the selected Activity Sequence(s) is loaded---Note: multiple sequences can be merged into 1 file
    1. ACE/Sequence Team assign deliver pass + priority if not previously assigned

    2. The Command Object is loaded to File Store….. Command Record's relation to Activity Record is identified
        1. A Non-interactive Command Object (NIPC/NIFL or Non-MGSS supported Mission Commands) enter at this point

8. Mission Controller/Sequence team can modify command file's selection data in the Command Record at any time.

IV. Command Delivery Service

Figure 5: Overview of the Command Delivery Service including its Accounting



Follow the numbers in the diagram above to understand the flow of accounting and tracing information end-to-end:

1) User communicates through Uplink Portal---Portal provides Security and Web Access for entire uplink process set
2) Portal provides read and write access to File Store –*Note: Operationally File Objects are read only, once loaded.*
3) Portal sends File Object headers, User Uplink and CDS directives/queries to Information Bus, Portal accepts images of requested Records, File Objects and pertinent event notices for delivery to User
4) Uplink Agent subscribes to information bus for File object headers, User directives and CDS, CRS and TLM events
    - Uplink Agent uses Database Mgt System; creating and updating Uplink Activity records as required
    - Uplink Agent will create logs and update the uplink archives at the end of the service instance
5) DISA provides a DBMS for the use by all Control and Accounting Agents which communicate via the Information Bus
6) DISA's DBMS uses DSMS File store to hold Uplink Request, Activity and Command Records.
7) SPS Agent Configures CDS at start of a service providing it with Pass ID for selection of Command Records
8) The CDS once configured is authorized to access the Command Records related to that pass.
9) CDS selects files for delivery using information in Command Records and accesses the specified file collection
10) CDS communicates with CRS using CCSDS SLE CLTU protocol
11) CRS delivers event notices to Information Bus (*only required for operations not involving JPL supplied CDS*)
12) TLM delivers CFDP response and status messages and Command Acknowledgement notices (TBD)
    - Command Acknowledgement notices should be standardized through the CCSDS
        - These can be delivered by the TLM services or by the user via the Uplink API

V. MRO Example

The MRO mission was the first high rate mission to incorporate CFDP for product delivery. The accounting process for this mission was started as a prototype for the development of a multi-mission service within JPL. Because of time criticalities the process has much less formalities and uses a paper trail for workflow and science product annotation processes. The MRO process is focused almost exclusively on product account: what products where planned and accounting for their acquisition. The science team would provide unique names for each of the products that their instruments where commanded to product. A listing of the products was generated from the sequence testing process to set the accounting system in motion. The spacecraft commands that were uplinked to the instruments provided the names for the products to be generated which would be included in the downlinking of the products. These products where transformed after creation into a series of CFDP Protocol Data Units (PDUs), that included the product's given name, and were then incorporated into the telemetry frames that would eventually be delivered to earth. These frames were stored in the spacecrafts mass store until a downlink tracking pass could transfer them to earth. The spacecraft produces a notification of the creation of the named product and transmits it in its engineering data. Typically the engineering data would be received before the actual product so upon receipt of the notification the ground would update the status of that product from "predicted" to "created". Once received the frames are delivered to JPL's CFDP processing entity the recreation of the product is stated and the accounting service is kept with the latest progress of the capture and product recreation process. When ever a product is published, sent to the science team, it is noted in the accounting service. Thus the accounting service will monitor the spacecraft product request, execution and delivery process from end to end. The user can query the system for a products status at any time. In the accounting service the times of receipt of notifications and CFDP PDUs are captured along with the time of delivery of the products to the science teams. So not only can the science teams follow the progress of their requested products but the ground services can determine where the products are and if the ground systems have met their obligations to the project for the delivery of the products in a timely manner.

VII.Conclusions

The major conclusions are: TBD

Acknowledgments

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

References