SPACECRAFT DATABASE STANDARDISATION: AN OMG INITIATIVE

Mario Merri[1], Roger Thompson[2], Janice Champion[3], Gerry Simon[4], Peter Shames[5], Mike Rackley[6], Jim Wetherbee[7]

[1] ESA/ESOC, Robert-Bosch-Strasse 5, D-64293 Darmstadt, Germany
Tel. +49 (6151) 902292, Fax +49 (6151) 903010, e-mail: Mario.Merri@esa.int

[2] Science Systems (Space) Ltd., Methuen Park, Chippenham, Wiltshire, SN14 0GB UK
roger.thompson@scisys.co.uk

[3] Boeing Satellite Systems, 2000 El Segundo Blvd, El Segundo, California 90245, USA
Tel. (310) 416-4544, Fax (310) 416-5188, e-mail: janice.a.champion@boeing.com

[4] Lockheed Martin, CERES, 720 Irwin Ave., Schriever AFB CO 80912, USA
Tel. +1 (719) 567-8349, e-mail: Gerry.Simon@schriever.af.mil

[5] JPL/CalTech, 4800 Oak Grove, MS 301-265, Pasadena, CA 91109 USA
Tel. +1 (818) 354-5740, Fax +1 (818) 393-1333, e-mail: Peter.Shames@jpl.nasa.gov

[6] NASA, GSFC, Greenbelt, MD, USA
Tel. +1 (301) 286-5528, Fax +1 (301) 286-5719, e-mail: Mike.Rackley@gsfc.nasa.gov

[7] OMG Space Domain Task Force, 250 First Ave. Suite 201, Needham, MA 02494, U.S.A.
Tel. +1 (781) 444 0404, Fax. +1 (781) 444 0320, e-mail: James_Wetherbee@omg.org

ABSTRACT

In July 2001, the Space Domain Task Force of the Object Management Group (OMG) issued a Request for Proposal (RFP) on Telemetric and Command Data Specification. This RFP solicited proposals that define telemetry and commanding data specifications in support of all phases of the satellite, payload, and ground segment lifecycle: system design, development, test, validation, and mission operations. In fact, telemetry and commanding data design is still often performed multiple times in multiple formats by multiple contractors during the lifecycle of the satellite, well before the satellite is ever deployed for mission operations. Standardisation of telemetry and commanding data descriptions for spacecraft health and safety monitoring, as well as payload interfaces, will reduce the cost of these implementations and decrease the development schedule, integration, and test of the satellite and its component systems. A common, adaptable specification can also be used to support multiple heterogeneous missions, facilitating interoperability among ground control systems, simulators, testing facilities, etc.

Three proposals were received by the OMG in response to the above RFP. These proposals were submitted by the European Space Agency (ESA), Boeing and Lockheed Martin. All proposals were based on the eXtensible Markup Language (XML) [9-12] but were embedded in the current "modus operandi" of their respective organisations. Despite the organizational and functional differences (packet telemetry vs TDM), it was decided to merge forces and to aim at a single submission that would cover the three approaches. Both NASA/GSFC and JPL supported this goal and agreed to actively contribute to this to ensure that NASA requirements were also addressed. This paper presents

the status of and plans for this initiative and briefly introduces the emerging standard specification of telemetry and commanding data definition as it currently stands.

OMG INTRODUCTION

The Object Management Group (OMG) [1] is an international not-for-profit software consortium that is setting standards in the area of distributed object computing. It is a vendor-neutral membership-driven organisation and has approximately 800 members who are working towards developing and refining these standards. Any company may join OMG and participate in the OMG standards-setting process.

In March of 1999, a Space working group was created in the OMG. The group became an OMG Domain Task Force (DTF) [2] in September of 2000. The group has issued two RFPs and a number of Requests for Information (RFI). The Space Domain Task Force's mission is to foster the emergence of cost effective, timely, commercially available and interoperable space, satellite and ground system domain software components through CORBA technology.

The Space DTF's goals include:

- Clarify space, satellite and ground system requirements
- Recommend technology for adoption that enables the interoperability and modularity of CORBA based space, satellite and ground system domain software components
- Encourage the development and use of CORBA based space, satellite and ground system domain software components, thereby growing the object technology market
- Encourage the use of UML to describe the architectures of their distributed systems in a standard way
- Encourage continued space industry member participation leverage existing OMG specifications.

To satisfy these goals the Space DTF will:

- Use the Object Management Group (OMG) technology adoption process to standardise interfaces for software components, services and frameworks in space applications
- Create a Space Architecture and Roadmap for the Space Industry world-wide
- Leverage existing OMG specifications
- Involve all interested members of the OMG in the OMG Space Domain Task Force
- Issue RFIs, RFPs and RFCs for CORBA-based technology relevant to space
- Identify relevant standards, architectures, research and technologies in space applications
- Assist and advise the Liaison Sub-Committee regarding its relationship with related Standards Organizations and Consortia
- Participate and present in other space industry consortium to encourage further OMG participation.

The Space DTF has concentrated its early efforts on developing a reference architecture [3] and on the standardization of the Telemetry and Commanding aspects of the Space Domain. These efforts include the RFP described in this paper (RFP-1) as well as a second Request for Proposal that will build off of this standard to include Monitor and Control Access [4] – the way in which certain telemetry and commanding information is passed through the overall systems (RFP-2).

The rest of this paper describes RFP-1 – the Data Specification of Commanding and Telemetry Data.

THE NEED FOR SPACECRAFT DATABASE STANDARDISATION

The OMG Space DTF conducted an Information Day in December of 2000 in which a number of presenters identified the need for a standard specification to define spacecraft telemetry and commanding data. This specification can be used to describe the spacecraft communication interfaces throughout its operational life and is important to overall spacecraft operability. Desire to produce this specification of the telemetry and commanding area resulted in the development and issuing of RFP-1 [5].

For a given mission there are a number of lifecycle phases that are supported by a variety of different systems and organizations. Additionally, many of these organizations support multiple heterogeneous missions using common ground segment infrastructure, which may be composed of possibly heterogeneous software systems. Telemetry and command definitions must be exchanged during all of these phases, among all of these systems and organizations. This is made difficult and costly because there is no standard method for exchanging this information. The lack of standardization currently requires custom translation and ingestion of the telemetry and commanding information at each step in the process. This customisation is inherently error-prone, resulting in the need to revalidate at each step in the lifecycle.

A typical example of this process is between the spacecraft manufacturer and spacecraft-operating organization. The spacecraft manufacturer defines the telemetry and command data in their own format which may be different than the one used in the ground segment. This creates the need for database translation, increased testing, software customization, and increased probability of error. Standardization of the command and telemetry data definition format will streamline the process, allowing dissimilar systems to communicate without the need for the development of mission specific database import/export tools.

Ideally, operation of a given spacecraft should be able to transition from one ground system to another by simply ingesting an already existing command and telemetry database that is compliant with this command and telemetry database specification. In addition, standardisation will enable space or ground segment simulators to more easily support multiple heterogeneous missions.

As a quick background on spacecraft telemetry and commanding, the following paragraphs will serve to educate those not familiar with spacecraft communications data.

Telemetering is defined (from IEEE Std 1000 [1972]) as “measurement with the aid of intermediate means that permit the measurement to be interpreted at a distance from the primary detector.” All measurements on board the spacecraft are transmitted to the ground system in a telemetry stream. Telemetry as used here refers to these measurements whether on-board the spacecraft or transmitted to the ground system. Most telemetry measurements will require engineering unit conversion and measurements will have associated validation ranges or lists of acceptable values. Telemetry data is typically associated with on-board sensors (e.g. temp, pressure, voltage), effectors (e.g. actuators, switches), or (sub-)system state (powered on, stand-by).

Commands, as defined for this RFP, are messages originating from the ground or the spacecraft to perform a function on the spacecraft or ground system. Spacecraft commanding usually implies coding and packaging of the command information, validation and verification, as well as authorization to perform. Commands are instructions to perform some (possibly detailed or high level) operations or actions.

Telemetry and Commanding data are necessarily related to one another, with some command information originating in, or derived from, telemetry and some commands relating to particular telemetry measurements. Therefore, the ability to relate individual telemetry with one another and to commands is a very important element of the dictionary definition. Packaging of both telemetry and commands can be performed in a number of ways. The most common way to package data for transmission is to use the CCSDS Telemetry and Commanding Packaging format, though TDM systems are also used to deliver similar command requests.

APPROACH TO RFP INTEGRATION

Three submissions were received in response to RFP-1 from three different organisations: the European Space Agency (ESA), Boeing and Lockheed Martin. Each submission describes in XML the current structure of their Telemetry and Command databases [8].

It was agreed within the Space DTF that the best way to proceed was to merge forces and to work towards a single, combined, response that would accommodate the approaches currently in place at each submitting organization. It was also agreed that NASA/GSFC and JPL, both members of the Space DTF, would contribute to this process in order to make the resulting single submission applicable to them as well. The idea behind this is that each organization would develop its own software tool that would allow converting from the generic XML-based format to its own specific one.

In order to achieve the above goal, the work was split as follows:

- Telemetry Database was taken by Boeing
- Command Database was taken by Lockheed Martin
- Packaging Database was taken by ESA
- Analysis and review was taken by NASA (GSFC & JPL)

The remainder of this paper briefly presents the results that have been achieved to date. These are just preliminary results that require further improvement and iteration. Only the UML Class Diagrams have been present here in the interest of brevity and clarity. The complete XML schema specification will be made available to anyone who is interested.

DATA DESCRIPTION CONTEXT

Two types of data structuring are represented in the proposal: Time Division Multiplexing (TDM) and Consultative Committee for Space Data Systems (CCSDS) frame and packet standards [6,7]. The proposal provides a merged format that will address the needs of processing both TDM and CCSDS packet streams.

TDM streams are uniquely characterized by the presence of a predictable and repeated telemetry format. This is because a spacecraft generating a TDM stream stores a table of sequencing codes consisting of ordered sets of telemetry requests. These sequences describe the order that spacecraft data will appear within periodic data segments (minor frames) and the ordering of minor frames within transmission frames (major frames). Since these major and minor frame sizes and sequencing code tables are known on the ground, the order in which telemetry requests are generated on the spacecraft and assembled into the telemetry stream can be precisely predicted. Commands within TDM streams are also converted into binary forms as required, assembled into fixed frames, and encoded for transmission to the spacecraft.

CCSDS streams are composed of dynamically ordered packets of information. CCSDS Packet Telemetry Data System defines two data structures -- SOURCE PACKETs and TRANSFER FRAMEs. The SOURCE PACKET is a data structure generated by an on-board APPLICATION PROCESS in a way that is responsive to the needs of that process. It can be generated at fixed or variable intervals and may be fixed or variable in length. The SOURCE PACKET PRIMARY HEADER contains an APPLICATION PROCESS IDENTIFIER used to route the packet to its destination. The header also carries information about the length, sequence, and other characteristics of the packet. An optional SOURCE PACKET SECONDARY HEADER is provided for standardised time-tagging of SOURCE PACKETs, and to carry application-unique ancillary data.

The CCSDS TRANSFER FRAME is a data structure that provides an envelope for transmitting packetised data over a noisy space-to-ground channel. It carries information in the TRANSFER FRAME PRIMARY HEADER that permits the ground system to route the TRANSFER FRAMEs to their intended destination. The TRANSFER FRAME is of fixed length (for a given PHYSICAL CHANNEL during a MISSION PHASE). The TRANSFER FRAME PRIMARY HEADER provides the necessary elements to allow the variable-length SOURCE PACKETs from a number of APPLICATION PROCESSes on a spacecraft to be multiplexed into a sequence of fixed-length frames. Short packets may be contained in a single frame, while longer ones may span two or more frames. Since a packet can begin or end at any place in a frame, the entire data field of every frame can be used to carry data; there is no need to tune the sizes of packets or their order of occurrence to fit the frames. CCSDS commanding uses a similar approach to constructing variable length frames that are encoded as required for transmission.

All of the processing that is required to accept RF signals, demodulate and decode the symbol stream, do frame synching, and extract minor frames or packets is expected to be performed by front end ground system equipment. It is often done in hardware, though it may use software components. The focus for this packaging approach is to describe the contents, structure and assembly of Telemetry and Command Data elements that are themselves composed of data items and structures of data items. The assumption is that the results of the respective telemetry frame synch processes will be delivered as messages and that the commands that are to be delivered will also be transferred as messages and assembled into transfer frames via some similar process Figure 1 shows a graphical representation of the relationships among the various data elements that may be derived from an RF signal.

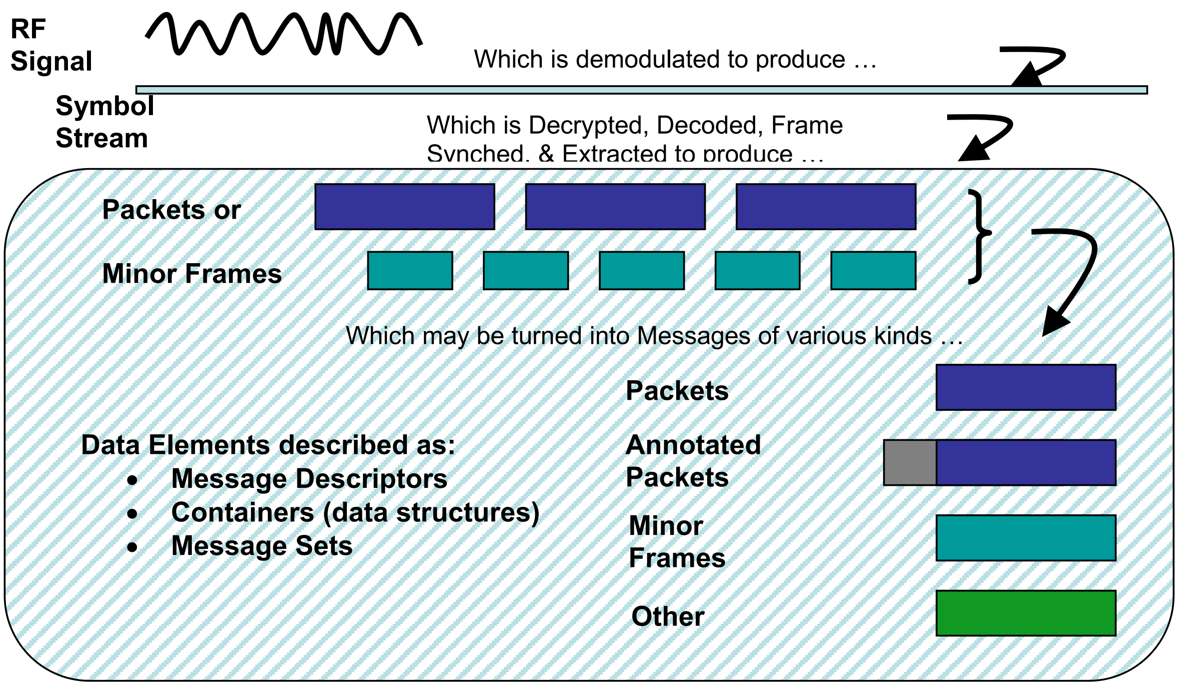

Figure 1: Data Description Context

PACKAGING DATABASE

In order to arrive at a generic schema that can support Telemetry and Command Data Specifications for the widest possible range of TM/TC protocols, the approach has been taken of separating the definition of generic data items (Telemetry Parameters and Symbolic Telecommands), from the packaging of these data items into the appropriate message formats supported by the particular transfer protocol used [TDM frames, CCSDS packets, or other proprietary formats].

The purpose of the packaging schema is to define both the structure of messages transferred on the space-ground interface and, where appropriate, their content in terms of parameter and command data items. The packaging schema recognises that not all data flowing on the space-ground interface corresponds to simple parameter and command data, but can include elements containing other data items (e.g. memory dumps) that are themselves outside the scope of the proposed spacecraft database standard. The schema also allows for these space-ground communication elements (packets or minor frames) to be embedded within ground segment messages supporting the protocol between ground station and control center.

The packaging schema is based around the concepts of messages and containers. A simple example might be a CCSDS frame received from a ground station, containing a single housekeeping TM packet. The message would define how to recognise the packet, the container would then define the structure of that packet. See Figure 2 for these UML class diagrams.

Message Descriptors define a generic structure only in terms of the information necessary to discriminate between a set of possible message types, which is essentially a description of an envelope and its addressing information. Each member of the set of possible message types is defined as a container that defines the detailed message structure. The message identifies a set of discriminator fields, while the container describes the valid value set for those fields. The container also defines its structure in terms of its constituent parameters.

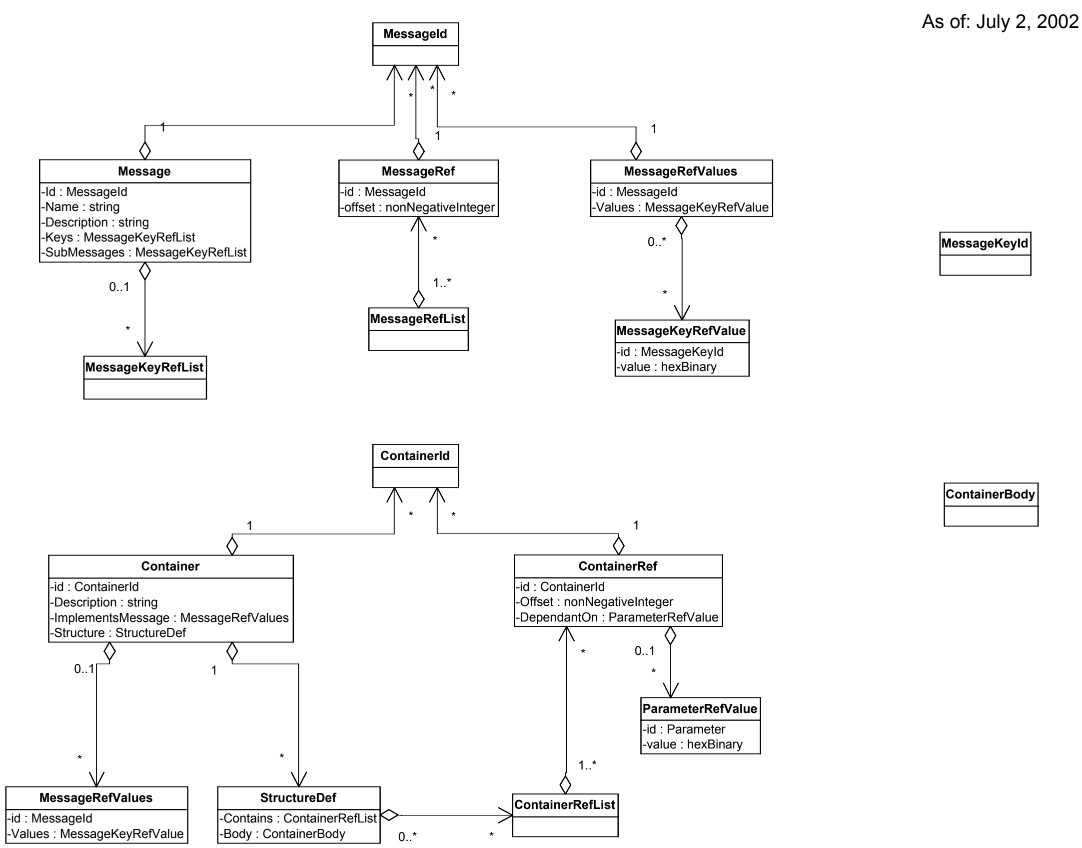

Figure 2: Message & Container Class Diagrams

Each message is defined as a set of message keys which give the bit offset relative to the start of the message and the width of the key. These keys are used to discriminate among different message types and may be used to route the messages for processing or other special handling. The contents of each message may be described as one or more containers, and/or other messages with their own key fields.

Containers (data structures) can be defined in terms of other containers and basic data elements, either parameters or command items. This recursive definition of messages in terms of other messages or container, and of containers in terms of other containers or data elements, permits definitions of arbitrary structures and allows the definition of data structures at any protocol level to be encapsulated.

Essentially, the packaging schema comprises the following components:

- Parameters: these correspond to the telemetry and command parameters [fields or arguments] that have been described previously.
- Containers: these define the structure of packets and frames, in terms of sub-containers and/or parameters.
- Messages: these describe a special container that defines key fields [Message Keys] that can be used to discriminate between different message formats. For example, this may be used to identify a specific packet or frame. Messages can contain sub-messages. Values of the Message Keys are associated with a specific container instance.

TELEMETRY DATABASE

A telemetry stream can be defined as a long stream of binary data originating on the spacecraft that contains health, status and other information generated by the spacecraft processors and payloads. Decommutation is defined as the process by which the telemetry stream is broken into component

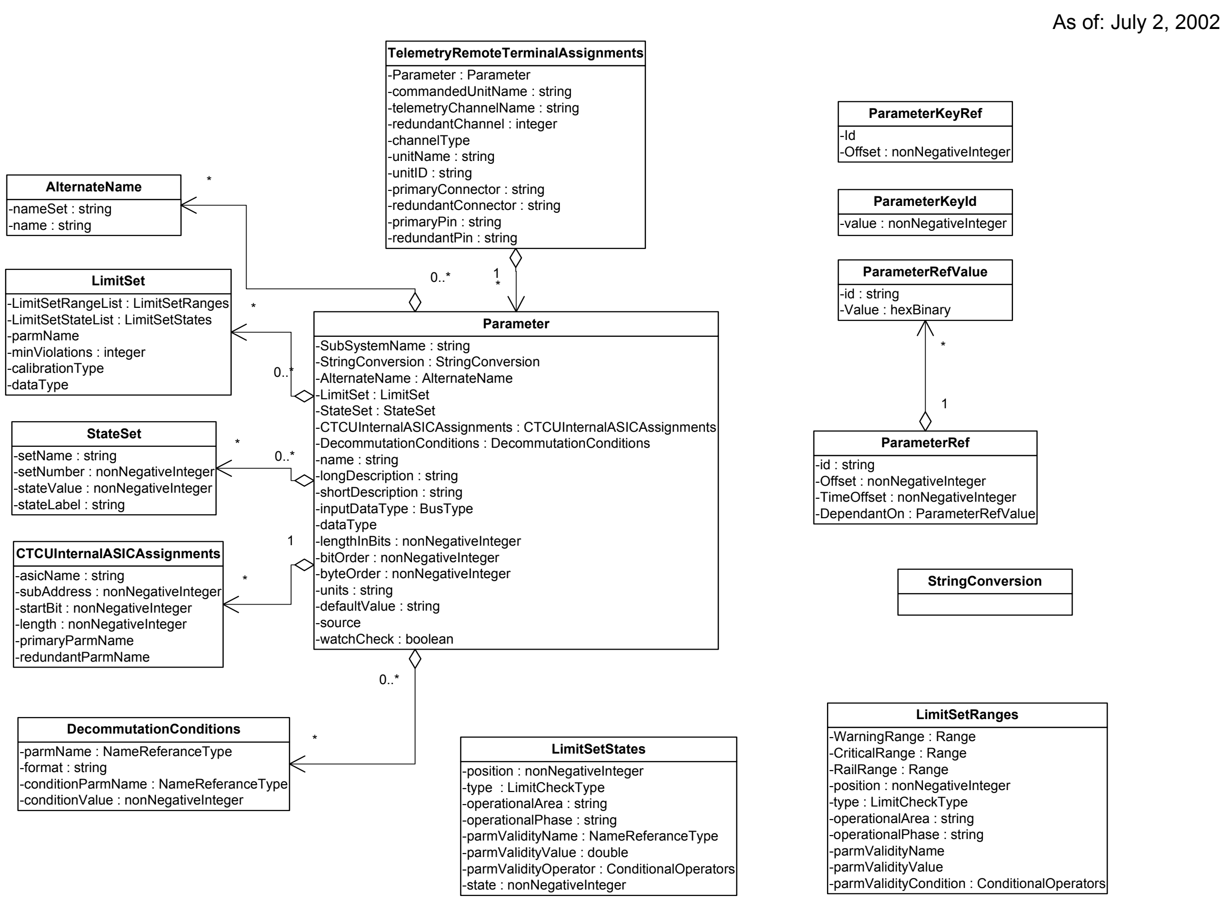

Figure 3: Telemetry Class Diagram

pieces for use by the ground software. The telemetry portion of the data exchange is devoted to providing information required to describe stream assembly on the spacecraft, to decommutate the telemetry stream from the spacecraft, and to provide the Engineering Unit (EU) conversion information used to display the telemetry stream to the operator.

The telemetry database contains data structures necessary to decommutate the data (see Figure 3 - Telemetry (Stream) Class Diagram). Sometimes special conditions exist where two identical records exist in the normal telemetry frame element with the exception of the governing parameter (or ground variable representing the telemetry element). These special conditions are defined in the DecommutationConditions structure.

The spacecraft data may need to undergo conversions or calibration, so there are structures to hold the information needed for these conversions and the associated algorithms. Telemetry may also require validation after decommuntation and/or conversion, so additional structures are provided for identifying invalid conditions (ParameterLimitSet.) There are other miscellaneous structures holding data necessary for other telemetry-related ground station operations.

COMMAND DATABASE

While the amount of command data in a space system is typically not as voluminous as telemetry data, it is especially important that command data be passed between organizations in a standardized way to eliminate the possibility of improper translation or interpretation. Spacecraft health and safety depends upon the correct transmission of only an appropriate set of commands.

One of the challenges faced in creating a command data format, stems from the rather imprecise language we in the industry use in regards to commanding. In our industry the term *command* actually has two related, but different meanings. The first meaning for *command* refers to a specific set of bits sent to a device (e.g., a spacecraft). The second meaning for *command* actually refers to a description of what a legal command is. A command database usually refers to the second meaning of command while the phrase "send a command" refers to the former meaning for command. This specification

As of: July 2, 2002

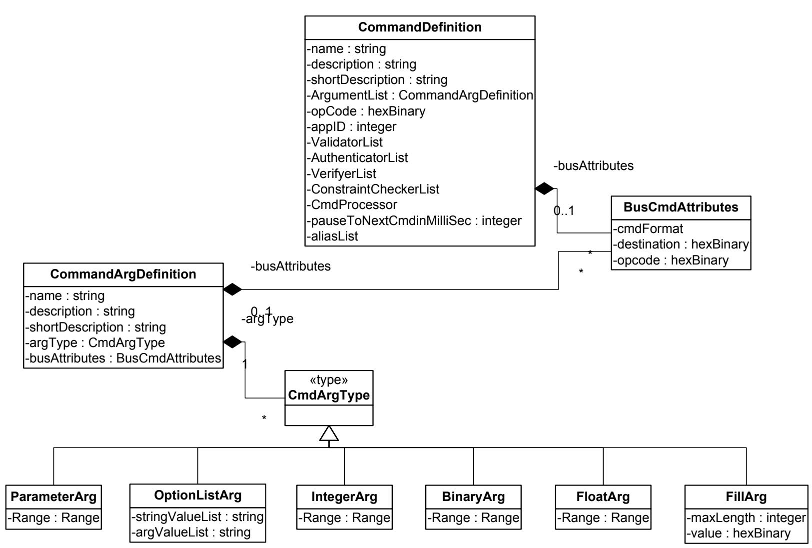

Figure 4: UML Diagram for Command Data

always uses the term *command* to refer to specific instructions sent to a space system and *command definition* to refer to a description of possible commands that could be sent to the space system. In Object Oriented parlance, a *Command* “has a” *Command Definition* and Space Systems “have” many *Command Definitions*.

At the simplest level, a command has an opcode and possibly some arguments. However, the detail required to specify a legal argument, specify the mechanisms used to turn human readable arguments into machine readable arguments, provide instructions to the ground system on how to process the command and to specify how to assemble and package the command, is quite complex. Figure 4 is a high level UML class diagram depicting the information the command definition database will store. This UML model has been converted into a W3C schema and inserted into the SpaceSystem schema which in turn defines the command data structure in XML.

CONCLUSIONS

The activity described in this paper has demonstrated that it is possible to define a single set of common command and telemetry database schema that are capable of describing a broad range of common spacecraft operating modes and communication link designs. The schema supports spacecraft that use a variety of different signalling and data structuring modes, including, at their most basic level, both packetized data (CCSDS compliant or otherwise) and time division multiplexed (TDM) data. In addition a variety of different commanding approaches are supported.

Beyond that, we have demonstrated that it is possible to use XML to define common schema that may be used for the exchange of command and telemetry databases among different ground data system elements. While some of the most recent systems use XML to define various space operations data items there has not been, to date, any attempt to define a generic schema that is suitable for this purpose. The eventual publication of this generic schema should enable the development of common approaches for the exchange of these data. Adoption of this approach is expected to have a significant impact on the cost of developing, validating, and exchanging these critical spacecraft information files.

ACKNOWLEDGMENTS

Thanks are given to the other members of the OMG Space Domain Task Force (Space DTF) who participated in the initial development of this overall approach and who have supported this activity during working meetings. We would each also like to acknowledge our individual organizations and sponsors who had the forethought to support this standards development effort.

REFERENCES

[1] Object Management Group, http://www.omg.org

[2] Space Domain Task Force, http://space.omg.org

[3] Space Domain reference Architecture, http://cgi.omg.org/cgi-bin/doc?space/

[4] Monitor/Control Data Access, http://cgi.omg.org/cgi-bin/doc?space/02-01-04

[5] Telemetric and Command Data Specification, Space RFP-1, 20 Aug 2001, Space/01-04-01

[6] CCSDS Packet Telemetry, CCSDS 102.0-B-4

[7] CCSDS Telecommand, CCSDS 203.0-B-1

[8] Telemetry and Telecommand Packet Utilisation, Draft 10, Apr 2002, ECSS-E-70-41

[9] W3C Recommendation - Extensible Markup Language (XML) 1.0 (Second Edition, 6 October 2000), http://www.w3.org/TR/REC-xml

[10] W3C Recommendation - XML Schema Part 0: Primer (2 May 2001), http://www.w3.org/TR/xmlschema-0/

[11] W3C Recommendation - XML Schema Part 1: Structures (2 May 2001), http://www.w3.org/TR/xmlschema-1/

[12] W3C Recommendation - XML Schema Part 2: Datatypes (2 May 2001), http://www.w3.org/TR/xmlschema-2/