Binary Representation of XML Infoset in the Space Domain Position Paper

Louis Reich, CSC/NASA
CCSDS Working Group Chair, XML Packaging

1. Background

1.1 Consultative Committee for Space Data Systems (CCSDS)

1.2 General CCSDS Organisation

In 1982 a number of the world's space agencies met to discuss problems common to space information and data systems. It had long been realized that the growing complexity of space missions as well as their associated costs could adversely impact space endeavors in the future unless specific efforts were undertaken to meet these concerns. Accordingly, the Consultative Committee for Space Data Systems (CCSDS) was established to perform end-to-end system analyses and to develop advanced solutions to these common problems.

The Committee's objective is to establish Recommendations in particular in those areas where interoperability between different space agencies is already, or is likely to become, important.

2. XML in the Space Domain Information Systems

2.1 What is special about Space Information?

Information derived from activities and observations in Space tend to be difficult and expensive to obtain and operations tend to be controlled remotely. This means that a good deal of telemetry, telecommand and digital transmission are required to get the data safely on the ground. Multi investigator and multi-national, publicly funded, missions are common. Observational data is in many cases made publicly available after a short time. Operational data is usually well curated, at least in the short term because of the need to trace faults in hardware, software or procedures. A great deal of information is therefore available for capture and there is good reason to store it systematically. However the data is derived from a multitude of interrelated sources and this fact can cause difficulties, especially after mission operations have ceased and mission specific software is no longer supported.

Figure 1 shows a schematic breakdown of the Space Domain by functional element. This shows the obvious components such as Spacecraft, Mission operations and various communications links. It also shows the relationship to some external Registry/Repository and the Science community, as well as hinting at the role that unique identifiers can play in the use of Space Information.

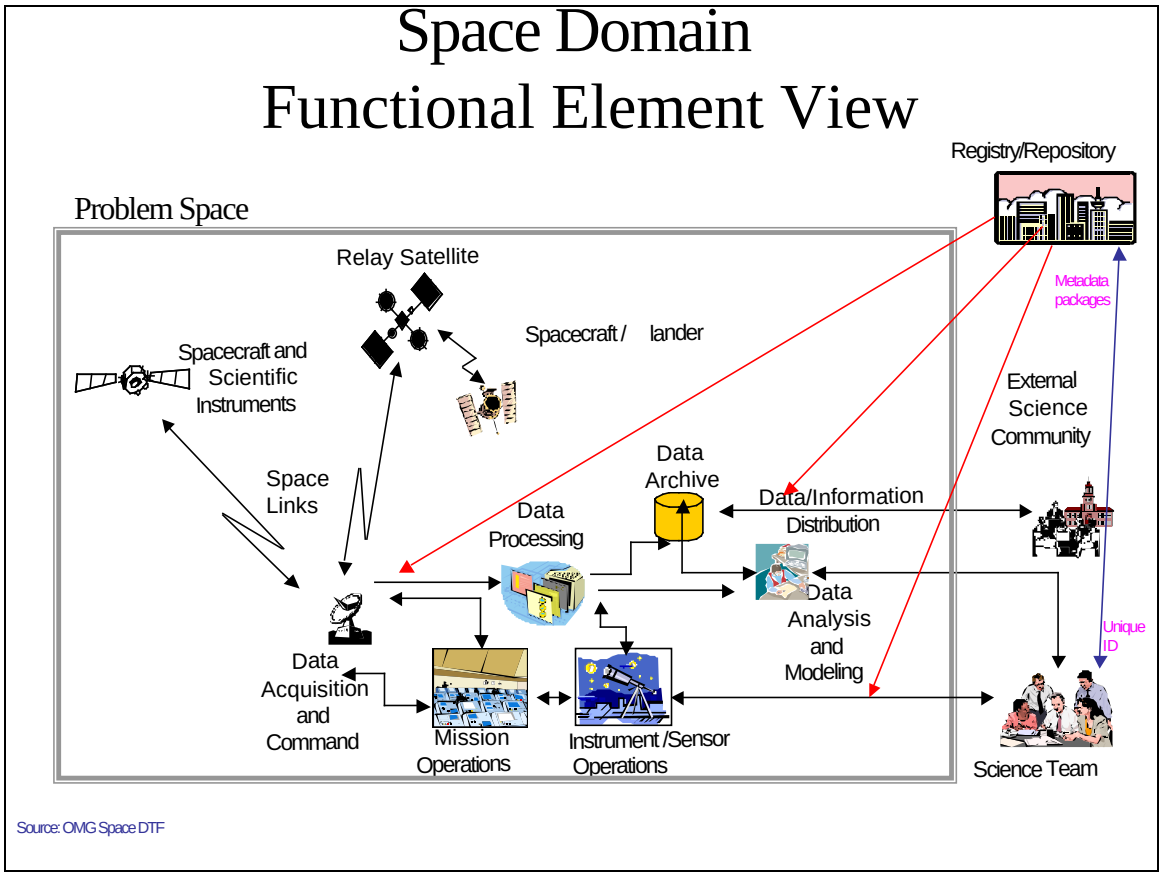

Figure 1

Currently the basic data transmitted in the Space Links tend to be binary, because of the limited bandwidth available However the Representation information (structure and Semantics) of that data stream could well be XML encoded. CCSDS has increasingly been developing higher-level standards involving XML. For example the Data Entity Dictionary Specification Language (DEDSL, reference 1 has a concrete syntax using DTD's (reference 2 and a concrete syntax using XML Schema has been drafted.

Figure 1 gives a view of the facilities during spacecraft operations, and so has no explicit locus for pre-mission design, development, or integration and test. . Each of these facilities houses a number of different producer and consumer elements, as well as the system users. The data flows between each of the elements represented by this diagram are primary targets for the application of XML. XML could, and probably would, be used within the elements, but the standards first intent would be to facilitate transfer of data between key producer and consumer elements. Figure 2 is an overlay, the green bubbles, showing a mapping of areas of potential XML usage to the space domain.

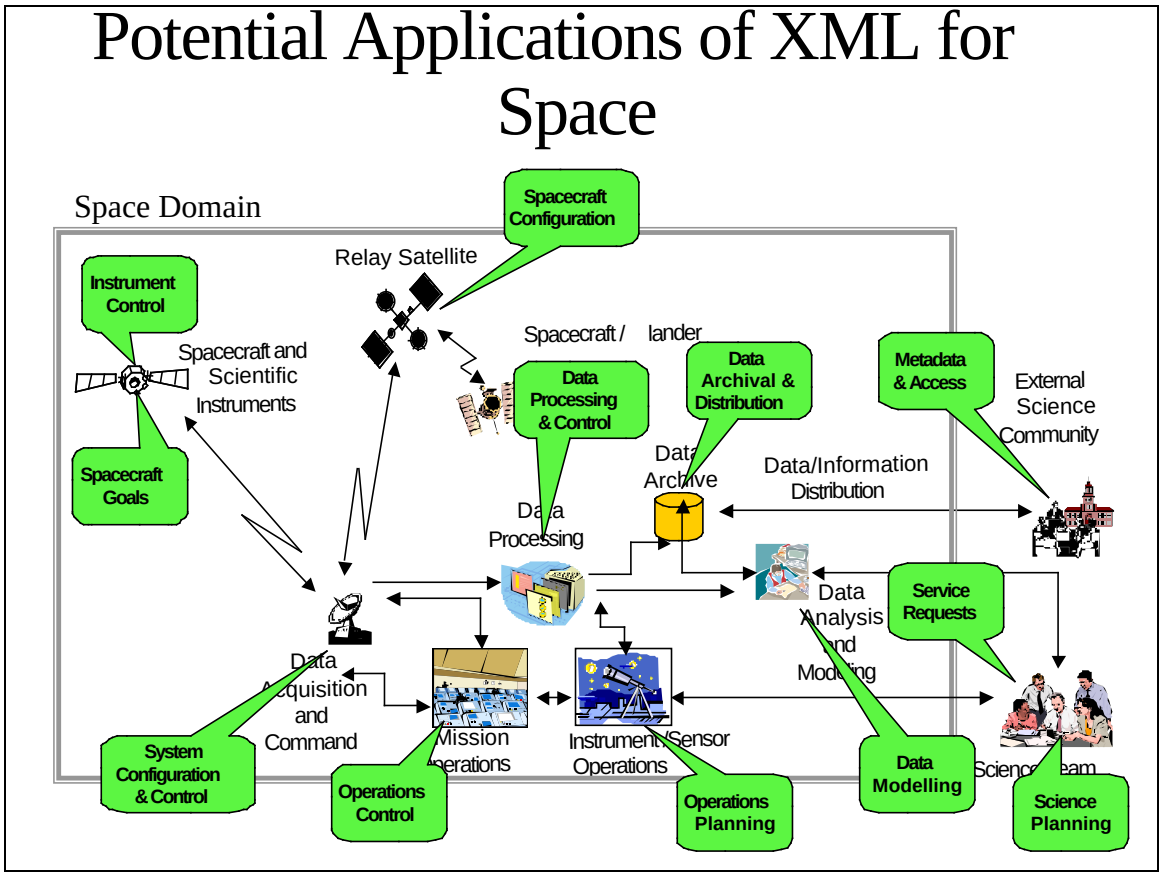

Figure 2

2.2 Why XML?

XML has become ubiquitous in just a few years and there is much effort going into new developments in many areas. It provides a vendor neutral way of sending messages between applications, which is to a large extent self-documenting. In the scientific arena in particular, the use of XML removes the need for many of the detailed discussions defining formats involving spaces and commas, instead allowing developers to focus on the semantic content rather than the minutiae of the format. Another reason for its rapid take-up is the availability of large amounts of COTS and Open Source software for dealing with XML.

2.3 Why not XML?

The datasets for many space science information users are stored in very large (tens of gigabytes) regularly structured binary files, often one or more large arrays or tables. They have tools for reading and manipulating these files, often written in languages like Fortran with primitive file handling capabilities. Whilst it is possible, in principle, to provide XML representations of such data it is not

clear why you would want to. An XML representation would have a number of drawbacks:

- The XML representation would be significantly (around 2-4 times) larger than the simple binary representation and therefore take longer to write, transport etc.
- Inappropriate representations: The proposed standard representation for a multidimensional array in XML to effectively build a tree of lists (everything in XML is a tree). This is a poor representation for scientific users because commonly required operations such as extracting a slice or a diagonal becomes difficult to do.

2.4 Current Usage of XML in the Space Domain

CCSDS has increasingly been developing higher-level standards involving the description of binary data using XML. CCSDS has study on a number of binary syntax description languages both in XML-based Data Description Languages (DDLs) such as HDX (reference 3), ESML (reference 4) and SML (reference 4) and in non-XML DDLs such as EAST (reference 5). We have also developed a concrete syntax Data Entity Dictionary Specification Language (DEDSL, reference [3]) using DTD's (reference [4]) and drafted a DEDSL concrete syntax using XML Schema.

The current focus of the XML effort is work on the use of XML for packaging scientific data, and producing Archive Information Packages using XML. These packages called XML Formatted Data Units (XFDU) enable the collection of the scientific data, engineering data, and operational data together with the representation metadata, the descriptive metadata, and other XML artifacts such as style sheets into a single object (file, message or document) described by a standard high level XML schema. A high level view of the XML Packaging Effort is included as Section 3 of this document. Work is also underway in the CCSDS Registry/Repository to provide preservation and access services so space data and metadata artifact.

3. XML Formatted Data Unit Overview

The XFDU package consists of a container that contains one, XFDU document and a set of byte-stream objects. There are three types of container object that will be supported.

- Archive formats (such as zip, jar or tar), which are already widely deployed, may be used as container.
- Message formats such as Soap with Attachments.
- The XFDU document can be considered as a container for ASCII/XML files or binary data encoded using XML Schema approved techniques.

There are seven sections that may appear in an XFDU document (i.e. Manifest). A high level XML Spy diagram of the XFDU is shown as Figure 3:

1. **Package Header** (packHeader): Administrative metadata for whole XFDUu such as version, operating system, hardware author etc and metadata about transformations and behaviours that must be understood

2. **Descriptive Metadata Section (**dmdSec): This section records all of the descriptive metadata for all items in the XFDU package. Multiple dmdSec elements are allowed so that descriptive metadata can be recorded for each separate item within the XFDU object. Descriptive information is intended for the use of Finding Aids such as Catalogs or Search Engines. DmdSec is specialization of **mdSec** type and can contain or reference desired metadata.

3. **Representation Metadata Section (repSec)**: Metadata sections based on OAIS RM Representation Information. The Representation Section and its subsections, syntax information (syntaxMd), static semantics (dedMd), and unclassified metadata (otherMd) are specializations of **mdSec**

4. **Preservation Description Metadata Section (pdiSec)**: Metadata sections based on OAIS RM Preservation Description Information, The subsections of the PDI Section - reference, context, provenance, and fixity - are specializations of the same base type mdSec

5. **Information Package Map Section (ipMapSec)** outlines a hierarchical structure for the original object being encoded, using a series of nested **contentUnit** elements. Content units contain pointers to the byte steam objects and to the metadata associated with those files.

6. **Data Object Section (dataObjectSec)** contains at list of dataObjEntry: A Data Object Entry contains the current file content and any required data to allow the information consumer to reverse any transformations and restore the file to the byte stream intended for the original designated community and describes by the Representation metadata

7. **Behavior Section (behaviorSec)** can be used to associate executable behaviors with content in the XFDU object. A behavior section has an interface definition element that represents an abstract definition of the set of behaviors represented by a particular behavior section. A behavior section also has a behavior mechanism that is a module of executable code that implements and runs the behaviors defined abstractly by the interface definition.

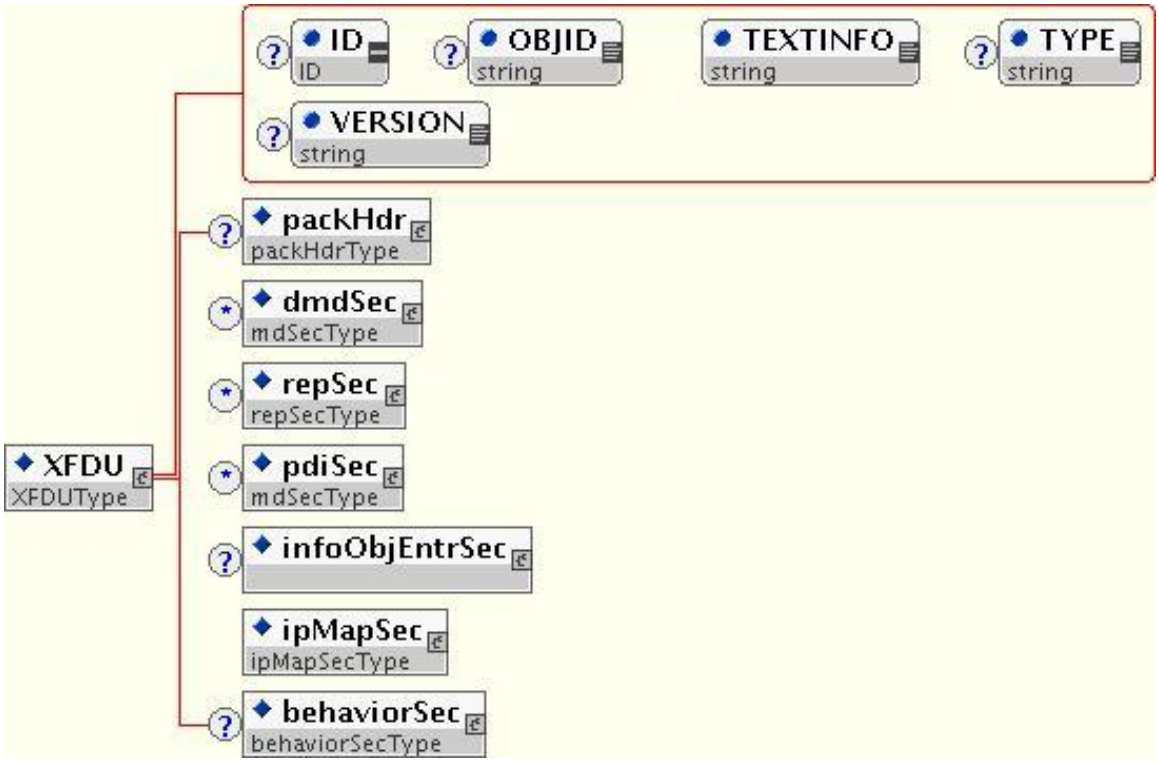

Figure 3

4. Answers to Requested Questions

1.*What work has your organization done in this area? (We are particularly interested in measurements!)*

- Currently we are investigating technical alternatives for packaging. We are evaluating alternative zip technologies such as Infozip for compression of scientific binary data We are also investigating the relative size of a zipped XML file with the binary base 64 encoded in the document versus an XFDU Package containing a pure XML manifest and files containing the “raw binary.
- We are also investigating techniques to allow a second level of packaging to contain the original package and a copy of the Package Header to allow a package consumer to understand the contents of the package with the overhead of unzipping the entire structure.

2. *What goals do you believe are most important in this area? (E.g. reducing bandwidth usage; reducing parse time; simplifying APIs or data structures, or other goals)*

- We believe all the candidate goals are important. Due to the binary downlinks and the size of some scientific datasets bandwidth reduction is the most obvious problem. However even with a substantial reduction, it is not clear that the Space Domain would use the binary XML unless data typing issues were resolved
- From an interoperability viewpoint the simplification of APIs and data structures by being able to integrate binary data with the current XML packages rather than needing to separate them into other objects or opaquely encode them would be an enormous benefit.
- It should be noted that simply providing an efficient coding of the current XML Schema simple types would not be adequate. A new set of binary types would be required

3. What sort of documents have you studied the most? (E.g. gigabyte-long relational database table dumps; 20-MByte telephone exchange repair manuals; 2 Kbytes web service requests)

- Our major problem is the several gigabyte scientific dataset and the bandwidth constrained downlink being fed by 600 megabit/sec instruments however there are many cases of smaller more structured documents or messages that would benefit.

4. *What sorts of applications did you have in mind?*

- See section 2

5. If you implemented something, how did you ensure that internationalization and accessibility were not compromised?

The space agencies currently have specified English/ASCII as the official language and we have not received any requests from our Asian members to go to UNICODE

6. How does your proposal differ from using grip on raw XML?

- See section 3

7. Does your solution work with any XML? How is it affected by choice of Schema language? (E.g. W3C XML Schema, DTD, Relax NG)

- Currently we can associate any syntactic description to the binary object. This can be extended to allow any XML Class definition

8. *How important to you are random access within a document, dynamic update and streaming, and how do you see a binary format as impacting these issues?*

- Random access and streaming are extremely important

5. References

[1] **CCSDS 647.1-B-1**: *Data Entity Dictionary Specification Language (DEDSL) - Abstract Syntax (CCSD0011)*. Blue Book. Issue 1. June 2001.
This has been adopted as ISO/DIS 21961.
<http://www.ccsds.org/documents/pdf/CCSDS-644.0-B-2.pdf>
[2] **CCSDS 647.3-B-1**: *Data Entity Dictionary Specification Language (DEDSL) - XML/DTD Syntax(CCSD0013)*. Blue Book. Issue 1. January 2002.
This has been adopted as ISO/DIS 22643.
<http://www.ccsds.org/documents/pdf/CCSDS-647.3-B-1.pdf>
[3] http://www.starlink.rl.ac.uk/ADAS2001/ADASS2001_giarettad.pdf
[4] http://esml.itsc.uah.edu/
[5] **CCSDS 644.0-B-2**: *The Data Description Language EAST Specification (CCSD0010)*. Blue Book. Issue 2. November 2000. This previous issue of this has been adopted as ISO 15889:2000.
<http://www.ccsds.org/documents/pdf/CCSDS-644.0-B-2.pdf>
[6] http://www.interfacecontrol.com/sml.asp