April 1, 2008

PDS3 Information Model Issues and Problems

PDS4 Data Model Working Group

-Draft-

History:

- First release March 21, 2008
- Edits in response full tech review, April 1, 2008

Reviews:

- Full tech review; Mar 27, 2008

Preface

Members of the PDS4 Data Model Working Group

Todd King (lead)
Steven Hughes
Ed Guinness
Lyle Huber
Chris Isbell

Table of Contents

1. Scope

This document captures issues or problems identified with the PDS3 information model. This document is written as a guide for the improvement of the PDS3 Information Model. The issues and problems are presented more as a checklist since solutions for each issue are not specified and will depend on the approach taken to improve the PDS3 Information Model.

2. Audience

The expected audience includes the following member of PDS:

- Managers and administrators
- Engineering Staff
- Technical Staff

or other individuals who wish to define the scope and extent of necessary changes to the PDS3 information model.

3. Introduction

The PDS4 Data Model Working Group has compiled a list of items that are considered to be issues or problems in the current PDS3 data model. The list encompasses problems with the data model, including the Planetary Science Data Dictionary, and standards defined in the Standards Reference. The intent was to document high-level issues and not to try to list every concern with individual keywords or data objects that the PDS standard group routinely works on. The items listed below attempt to describe the problems. In general recommendations for fixes are not given, although it was sometimes hard not to include such suggestions. The material presented here could be used to evolve the PDS3 data model or to help develop requirements for the PDS4 data model. Also, the list could be used to test whether the PDS4 data model has corrected the problems in PDS3 data model.

The material was derived from several sources. First, there was a Geosciences Node presentation to the PDS technical group that focused on issues that the node thought were important to address in modernizing the PDS data model. That presentation was following by the PDS Standards Working Group compiling a list of standards problems that should be addressed in the development of the next version of PDS standards as opposed to revisions done in the normal "Standards Change Request" process. Some material was also derived from the PDS4 discussion at the August 2007 PDS MC meeting. Finally, Steve Hughes also provided information that was partially based on these first two sources, but that had issues sorted into categories such as problems with the data model, data dictionary, and standards.

4. Issues and Problems

The issues and problems have been divided into two areas. The first area is related to issues and problems pertaining to the data model and its description in the data dictionary. The second area is related to issues and problems related to the standards reference. Generally the items are either requests for more flexibility and capabilities, requests for more rigor and less ambiguity, and requests for stronger enforcement. The issues and problems are presented as a concise list,

indented text below an issue provides a commentary on the issue or describes the context in which the issue arises.

4.1. Data Model and Data Dictionary

- Object definitions permit any term in the PSDD to be an optional element. The definitions should be rigorous and prescriptive (and not descriptive).

  The addition of the PSDD term to all objects occurred about 5 years ago and was one possible interpretation of the standards. Allowing any dictionary term in an object as an optional element defocused the object definitions.

- The relationships among model components are incomplete. Allowed cardinality is not specified for objects, also inheritance or sub-class relationships are not specified. In addition, some associations are described in narratives or by example. A more complete specification will reduce any ambiguities.

  Without specifying cardinality and sub-classing of objects translating from an logical model information model to a physical data model requires external interpretations or assertions.

- The current data dictionary model does not allow for the specification of the number of values allowed for a keyword.

  Without specifying cardinality of elements and objects translating from a logical model information model to a physical data model requires external interpretations or assertions. For example, relational schemas to represent one-to-one and one-to-many relationships are quite different. A risk with setting cardinality for elements is that the limits may be considered "arbitrary".

- It is difficult to promote data system interoperability among different data systems and agencies (ESA) or to promote cross-mission, cross-instrument search and data recovery because the specification of the standards or requirements for metadata is incomplete.

  This is closely related to the previous (cardinality) issue. To enable interoperability between independently operated systems the information model specification must be complete and unambiguious.

- There has been an inconsistent interpretation and application of the definition of a product. The definition of a product needs to be stated emphatically and enforced.

  In Chapter 4 of the Standards Reference, the first sentence defines a "product". This definition has not been universally applied.

Changing the definition will not, by itself, lead to stricter enforcement.

- The use of an implicit file object is not handled well. The file object should be explicitly defined in all cases so that required keywords can be defined (and validated).

  The implicit file object has not been treated in the same way as an explicit file object. This has lead to some confusion. In practice the implicit file object is different since content is less contrained than with the explicit file object.

- Products consisting of multiple files (compound products) are either poorly supported or impossible to describe. For example, a product consisting of multiple files organized in a directory tree cannot be described because paths are not allowed in pointers.

  Some storage structures may not be optimal for archiving. Currently a flat storage space is assumed. The issue here is whether hierarchical storage structures should be permitted and which ones should be allowed for archiving. It needs to recognized that the rules for archiving may be sub-optimum for access and vice versa.

- All targets (planets, satellites, small bodies, etc.) are treated as a single category. Because of the large number of targets the use of the list is cumbersome for providers. Also, some targets which are different in class (planet, satellite, small body) share the same name. Other organizations also have standard names for bodies.

  Examples of targets that share names include Halley (comet) and Halley (asteroid) or Amalthea (asteroid) and Amalthea (moon of Jupiter). The Small Bodies Node has defined a set of formation rule for creating target names eliminate ambiguities, but this method is not universally applied in PDS, hampering the use of the target name in locating related resources.

- There is no mechanism to describe observed regions such as fixed geographic areas or features in atmospheres.

  The GAZETEER object has the potential to address this issue, but the object definition refers to the " PDS Data Preparation Workbook" for the specification of the GAZETEER table record format. This document is not longer available from PDS.

- The situation where an instrument may be assigned to multiple instrument hosts (observatories) is not supported. The one-to-many instrument-to-instrument host problem needs to be refined.

These possibilities have been addressed in the past, but generally in an ad hoc way. A better procedure is desirable, but it will require redefining both INSTRUMENT and HOST. For example, is the telescope connected to a movable spectrometer part of the INSTRUMENT or part of the HOST? Past practice has made it part of the HOST; but that seems a poor logical choice.

- The radio science instrument with components on both the ground and on a spacecraft is poorly modeled.

  Though, radio science instruments have been described, even with the existing model.

- The standard values for some keywords have been poorly controlled. For example, there are currently several standard values of instrument_type that identify a magnetometer, which hinders using this keyword in searches.

  This is an enforcement problem, not one intrinsic to the standards. However, standard values are currently part of the standard data dictionary, so rectifying the situation requires a clear delineation (new version number)

- The attributes associated with the contents of the archive are included in the data dictionary. Currently attributes like volume_id, volume_set, and dataset_id have standard value lists derived from the current PDS holdings. This requires the data model (data dictionary) to be updated with each new addition to the archive.

  This issue is closely related previous (poorly controlled standard value list) issue. Separation of dictionary and standard value lists is needed.

- The PSDD is monolithic. All terms used in the archive are included in a single data dictionary. The PSDD should be divided into complementary dictionaries. One example is that there are 27 keywords with the term 'temperature' in the name, some of which apply to specific instruments or missions. Another example is that there are 20 keywords with 'latitude' in the name. The multitude of similar keywords makes it difficult for users to select keywords and to generate consistent labels.

  Local data dictionaries (currently allowed in PDS) help to resolve the "monolithic" issue. However, current operational practices are to merge all dictionaries (local and PDS) into a single PSDD. A transition away from the practice is underway.

- Aliases or synonyms in the data dictionary add to user confusion, e.g., instrument_host_id vs. spacecraft_id.

The INSTRUMENT_HOST_ID is defined to be "either a spacecraft or an earth base(d)" observatory or laboratory. The SPACECRAFT_ID is "synonym or mnemonic for the name of the spacecraft" and is treated as an alias for DSN_SPACECRAFT_NUM by some missions (AMMOS).

- There is no data dictionary versioning used in the data model. There is no keyword in the dictionary that can be used to indicate which version of the dictionary was used in creating an archive.

  The "pdsdd.full" file contains a generation date embedded in comments at the beginning of the file, but if the file is regenerated (even for the same version) this date will change. So, the date can not reliably be mapped to a version number. There is a "version" line also in the comments, but the value is always "OPS".

- The "data object description" and "pointer" relationship is somewhat disjoint.

  The "pointer" in labels has two usages which have a significantly different application. There are the "include" pointer forms and "data location" pointer form. The "data pointer" form maps data to object descriptions whereas the "include" forms allow objects or element values to be stored in separate (external) files. With the "data location" pointer an object definition must exist which has the same name as the pointer. The pointer could be moved into the object so that the two are tightly coupled and naming requirements can be eliminated.

- The current model does not adequately handle SPICE kernel data formats.

- The software objects are not completely modeled.

  The SOFTWARE object does not allow multi-file software "distributions" to be described.

- The document objects are not completely modeled.

  The DOCUMENT object allows a document to be "made up of one or many files in a single format", but documents composed of multiple files do not always have a single format for all files. For example, images included in a document are a different format than is used for the text portion of a document. Also, each representation (text, PDF, etc.) of a document requires a separate DOCUMENT object with information repeated for each instance. The standards are unclear on how multi-file documents are to stored and referenced.

- Catalog objects formatted as PDS labels or label fragments are not necessarily the most user-friendly way to deliver documentation to users from the archives.

  The PDS label was designed to be both human- and machine-readable. This has become common practice and the current industry standard is to use XML. (Which is less human readable than a PDS label). XML is typically transformed into a "user friendly" presentation. PDS labels could also be transformed for user viewing. Lack of user viewing capabilities does not mean the model is defective.

- Arbitrary limitations on the number of characters allowed for some keyword values requires an update to the data dictionary each time a new value exceeds the current size limit. Increasing and standardizing the size of certain classes of keywords such as *_id and *_name would reduce the number of data dictionary updates.

  Size (or length) limits are necessary in many instances. Some limits may have been imposed because implementation constraints. In some cases these constraints no longer exist. Since the reasons for some limits may be archaic, a re-assessment of current limits is warranted.

- The current data model does not account for dependencies between keywords.

  For example, the presence of certain optional keywords may require the presence of other keywords (bands, band_sequence, …).

- Individual products are not easily relocated. The interpretation of some attributes in a product description is dependent on an external file system organization or the dataset/volume context. This makes it difficult (or impossible) to form new collections (possibly based on a user's selection) consisting of portions of existing collections.

  The "volume" structure which requires different types of products to be stored in fixed locations (documents in "document" folder, data in the "data" folder) can result in file name conflicts when products from multiple "volumes" (or datasets) are combined into a new collection.

  This is an issue more for the delivery of archived products, rather than for the archive itself. An "archive system" manages and stores products and collections "as delivered" to the system.

- The list of data types which can be assigned to a dictionary element is incomplete.

Some elements use a data type and then specify limitations or extensions to the data type definition through a narrative in the standards reference. This has lead to differing interpretations of the standard. Data types and any constraints should be an integral part of the data dictionary to permit consistent application.

A possibly related issue is that the Data Dictionary characterizes keywords as being CHARACTER, REAL, INTEGER, etc. But the standard values for the keyword DATA_TYPE cover a much wider range. In fact the data types REAL and INTEGER in the Data Dictionary should be ASCII_REAL and ASCII_INTEGER based on the DATA_TYPE values.

4.2. Standards Reference

- Standards are unclear as to requirements versus recommendations; have not separated requirements from recommendations (e.g. and standards from policies). There is no clear set of requirements for archives.

  This is mostly an editorial issue with respect to the contents of the standards reference document. It would be preferable to have all policies clearly differentiated from the standards.

- There is no identification of acceptable or preferred data formats and derived object classes.

  All data objects are currently treated equally, but some shouldn't be permitted. The number of acceptable formats should be reduced or limited.

- The current standards need to better handle compound products. The handling of RECORD_TYPE definitions for files containing multiple objects of different types should be better defined.

  Similarly, DATA_OBJECT_TYPE is a nonsensical required keyword in DATA_SET_INFORMATION except for completely homogeneous data sets.

- Acceptable document formats (text vs PDF vs other?) needs to be updated.

  Most documents produced today are richly formatted and contain images to illustrate essential information. The required "text" format for documents is either difficult to product from source documents or results in a significant loss of information content.

- There are standards based on archaic hardware / software - (e.g. 80 character per line limit, use of RECORD_TYPE and RECORD_BYTES, case, underscores, etc.).

- The role of archive volumes needs to be re-evaluated in response to data sets primarily being on-line and distributed to uses electronically.

- The standards do not support characters with diacritical marks (e.g., accented and non-English characters), which are important for some international missions and archive partners.

  There is useful information in A.16.5 in the Standards Reference; whether this constitutes adequate support could be debated since it applies strictly only to GAZETEER_TABLE. There is a much broader question of how to handle non-Roman characters.

5. Conclusions

An attempt was made to capture all the known issues and problems with the PDS3 Information Model. Even so, this list should not be considered a complete list. Other weaknesses or desired improvements could be revealed through use cases or user scenarios. These issues and problems can be used both for improving the existing PDS3 Information Model and as one source of input for the definition of a next generation information model.