

Software/Document Overview

PDS 2010 Tech Session

June 10, 2009

PDS 2010 Data Design WG

Software/Document Similarities

- **Several discreet parts constitute a whole**
- **Application-specific formats**
- **General search should yield access to all formats**

In PDS4

- In PDS4, software and documents are digital data products on a par with observational data.
 - Unique identifiers
 - Associations with other PDS4 elements
 - Metadata for indexing and packaging

PDS4 SOFTWARE_SET Class

OBJECT = SOFTWARE_SET	Req/Opt	Cardinality
OBJECT = TAGGED_SOFTWARE_FORMAT	Required	1..*
DESCRIPTION	Optional	1
OBJECT = TAGGED_FILE	Required	1..*
LOCAL_IDENTIFIER	Required	1
CHECKSUM	Required	1
FILE_SIZE	Required	1
FILE_SPECIFICATION_NAME	Required	1
FILE_TYPE	Required	1
END_OBJECT = TAGGED_FILE		
OBJECT = SOFTWARE_BINARY	Optional	1..*
...		
END_OBJECT = SOFTWARE_BINARY		
OBJECT = SOFTWARE_SCRIPT	Optional	1..*
...		
END_OBJECT = SOFTWARE_SCRIPT		
OBJECT = SOFTWARE_SOURCE	Optional	1..*
...		
END_OBJECT = SOFTWARE_SOURCE		
END_OBJECT = TAGGED_SOFTWARE_FORMAT		
OBJECT = SOFTWARE_SET_DESC	Required	1
...		
END_OBJECT = SOFTWARE_SET_DESC		
OBJECT = IDENTIFICATION_SECTION	Required	1
URN	Required	1
IDENTIFIER	Required	1
TITLE	Required	1
VERSION	Required	1
DD_VERSION_ID	Required	1
PDS_VERSION_ID	Required	1
LABEL_REVISION_NOTE	Required	1
PRODUCT_CREATION_TIME	Required	1
END_OBJECT = IDENTIFICATION_SECTION		
END_OBJECT = SOFTWARE_SET		

SOFTWARE_SET references Documents

- The PDS4 SOFTWARE_SET class includes references to documents that assist in the description of the software “pieces / fragments”
 - The PROGRAMMERS_MANUAL_IDENTIFIER and the USERS_MANUAL_IDENTIFIER are each unique identifiers that reference documents “stored” somewhere within the PDS

OBJECT = SOFTWARE_SET_DESC	Required	1
VERSION	Required	1
AUTHOR_LIST	Required	1
DESCRIPTION	Required	1
NAME	Required	1
SOFTWARE_ID	Required	1
SOFTWARE_TYPE	Required	1
PROGRAMMERS_MANUAL_IDENTIFIER	Required	1
USERS_MANUAL_IDENTIFIER	Required	1
END_OBJECT = SOFTWARE_SET_DESC		

PDS4 DOCUMENT_SET Class

	Req/Opt	Cardinality
OBJECT = DOCUMENT_SET		
OBJECT = TAGGED_DOCUMENT_FORMAT	Required	1..*
DESCRIPTION	Optional	1
OBJECT = TAGGED_FILE	Required	1..*
LOCAL_IDENTIFIER	Required	1
CHECKSUM	Required	1
FILE_SIZE	Required	1
FILE_NAME	Required	1
FILE_TYPE	Required	1
END_OBJECT = TAGGED_FILE		
OBJECT = DOCUMENT_FORMAT	Required	1
DESCRIPTION	Optional	1
FORMAT_TYPE	Required	1
END_OBJECT = DOCUMENT_FORMAT		
END_OBJECT = TAGGED_DOCUMENT_FORMAT		
OBJECT = DOCUMENT_SET_DESC	Required	1
...		
END_OBJECT = DOCUMENT_SET_DESC		
OBJECT = IDENTIFICATION_SECTION	Required	1
URN	Required	1
IDENTIFIER	Required	1
TITLE	Required	1
VERSION	Required	1
DD_VERSION_ID	Required	1
PDS_VERSION_ID	Required	1
LABEL_REVISION_NOTE	Required	1
PRODUCT_CREATION_TIME	Required	1
END_OBJECT = IDENTIFICATION_SECTION		
END_OBJECT = DOCUMENT_SET		

Backup

PDS3 SOFTWARE Catalog File

```
PDS_VERSION_ID      = PDS3
RECORD_TYPE         = STREAM

OBJECT              = SOFTWARE
  SOFTWARE_ID        = SBN_REFSORT
  SOFTWARE_VERSION_ID = "V1.0"
  SOFTWARE_PURPOSE   = VERIFICATION

OBJECT              = SOFTWARE_INFORMATION
  SOFTWARE_NAME      = REFSORT
  DATA_FORMAT       = PDS
  SOFTWARE_LICENSE_TYPE = "PUBLIC DOMAIN"
  TECHNICAL_SUPPORT_TYPE = FULL
  REQUIRED_STORAGE_BYTES = 5000
  PDS_USER_ID        = "N/A"
  NODE_ID            = SBN
  SOFTWARE_DESC      = "This Perl utility sorts a file
  containing

  REFERENCE          a series of (properly-formatted)
                     objects according to the value of the
                     REFERENCE_KEY_ID elements. "

END_OBJECT = SOFTWARE_INFORMATION

OBJECT              = SOFTWARE_ONLINE
  ON_LINE_IDENTIFICATION = "http://pdssbn.astro.umd.edu/software/
  refsor.tar"
  ON_LINE_NAME          = "SBN REFERENCE Object Source Routine"
  NODE_ID               = SBN
  PROTOCOL_TYPE         = URL
  PLATFORM              = MULTIPLE
END_OBJECT = SOFTWARE_ONLINE

END_OBJECT = SOFTWARE
```

END

SOFTWARE Example

- The simplest software product consists of a single script file, a user's document and the PDS label:

```
SOFTWARE/SORTREF.LBL  
SOFTWARE/SORTREF.PL  
SOFTWARE/SORTREF.L
```

PDS3 to PDS4 Parallelisms

- In June/1999, Version 3.3 of the Standards Reference re-purposed the original SOFTWARE_INVENTORY template to become the SOFTWARE template
 - The SOFTWARE template was a more streamlined version of the SOFTWARE_INVENTORY
 - The SOFTWARE template provides information relevant to obtaining the software from online source(s)
- PDS4 proposes to re-purpose the PDS3 SOFTWARE template to be a data object description that treats software like “science data” where each is defined by a set of attributes
- PDS4 proposes to provide a “linking” mechanism whereby data providers can reference documents that assist in the description of the software “pieces / fragments”

SOFTWARE in PDS3

- In PDS3, the SOFTWARE metadata was provided in the form of a catalog template
 - Provides general information about “software” included in a PDS archive volume
 - Includes applications, tools, or libraries proven useful for the display, analysis, formatting, transformation, or preparation of science data

OBJECT = SOFTWARE_SET

Req/Opt Cardinality

OBJECT = TAGGED_SOFTWARE_FORMAT	Required	1..*
DESCRIPTION	Optional	1

OBJECT = TAGGED_FILE	Required	1..*
LOCAL_IDENTIFIER	Required	1
CHECKSUM	Required	1
FILE_SIZE	Required	1
FILE_SPECIFICATION_NAME	Required	1
FILE_TYPE	Required	1
END_OBJECT = TAGGED_FILE		

OBJECT = SOFTWARE_BINARY	Optional	1..*
OS_VERSION	Required	1
FILES	Required	1
SUPPORTED_OS	Required	1
SUPPORTED_ARCHITECTURE	Required	1
SW_FORMAT_TYPE	Required	1
SYSTEM_REQUIREMENTS	Required	1
PROGRAM_NOTES_IDENTIFIER	Required	1
END_OBJECT = SOFTWARE_BINARY		

OBJECT = SOFTWARE_SCRIPT	Optional	1..*
FILES	Required	1
INSTALL_NOTES	Required	1
SUPPORTED_ARCHITECTURE	Required	1
SYSTEM_REQUIREMENTS	Required	1
END_OBJECT = SOFTWARE_SCRIPT		

OBJECT = SOFTWARE_SOURCE	Optional	1..*
OS_VERSION	Required	1
COMPILE_NOTES	Required	1
FILES	Required	1
SOFTWARE_DIALECT	Required	1
SOFTWARE_LANGUAGE	Required	1
SUPPORTED_OS	Required	1
SUPPORTED_ARCHITECTURE	Required	1
SW_FORMAT_TYPE	Required	1
SYSTEM_REQUIREMENTS	Required	1
PROGRAM_NOTES_IDENTIFIER	Required	1
END_OBJECT = SOFTWARE_SOURCE		

END_OBJECT = TAGGED_SOFTWARE_FORMAT

OBJECT = SOFTWARE_SET_DESC	Required	1
VERSION	Required	1
AUTHOR_LIST	Required	1
DESCRIPTION	Required	1
NAME	Required	1
SOFTWARE_ID	Required	1
SOFTWARE_TYPE	Required	1
PROGRAMMERS_MANUAL_IDENTIFIER	Required	1
USERS_MANUAL_IDENTIFIER	Required	1
END_OBJECT = SOFTWARE_SET_DESC		

OBJECT = IDENTIFICATION_SECTION	Required	1
URN	Required	1
IDENTIFIER	Required	1
TITLE	Required	1
VERSION	Required	1
DD_VERSION_ID	Required	1
PDS_VERSION_ID	Required	1
LABEL_REVISION_NOTE	Required	1
PRODUCT_CREATION_TIME	Required	1
END_OBJECT = IDENTIFICATION_SECTION		

END_OBJECT = SOFTWARE_SET

OBJECT	Req/Opt	Cardinality
OBJECT = DOCUMENT_SET		
OBJECT = TAGGED_DOCUMENT_FORMAT	Required	1..*
DESCRIPTION	Optional	1
OBJECT = TAGGED_FILE	Required	1..*
LOCAL_IDENTIFIER	Required	1
CHECKSUM	Required	1
FILE_SIZE	Required	1
FILE_NAME	Required	1
FILE_TYPE	Required	1
END_OBJECT = TAGGED_FILE		
OBJECT = DOCUMENT_FORMAT	Required	1
DESCRIPTION	Optional	1
FORMAT_TYPE	Required	1
END_OBJECT = DOCUMENT_FORMAT		
END_OBJECT = TAGGED_DOCUMENT_FORMAT		
OBJECT = DOCUMENT_SET_DESC	Required	1
TITLE	Required	1
ACKNOWLEDGEMENT	Optional	1
AUTHOR_LIST	Required	1
DESCRIPTION	Optional	1
DOI	Optional	1
PUBLICATION_DATE	Required	1
RIGHTS	Required	1
END_OBJECT = DOCUMENT_SET_DESC		
OBJECT = IDENTIFICATION_SECTION	Required	1
URN	Required	1
IDENTIFIER	Required	1
TITLE	Required	1
VERSION	Required	1
DD_VERSION_ID	Required	1
PDS_VERSION_ID	Required	1
LABEL_REVISION_NOTE	Required	1
PRODUCT_CREATION_TIME	Required	1
END_OBJECT = IDENTIFICATION_SECTION		
END_OBJECT = DOCUMENT_SET		

Terminology

- A Data Model defines the entities to be processed, their **attributes**, and the relationships that add meaning.
- An **attribute** has alternate names.
 - Data Element
 - Vocabulary Term
- The set of all **attributes** in a data model is also called its vocabulary and is collected into a **data dictionary**.
- When defining an **attribute**, a set of **meta-attributes** or “attributes about attributes” are used.
 - The *name* of an attribute is a meta-attribute.
 - For example when defining the data element *sample_type*, the meta-attribute, *attribute_name*, has the value “sample_type”.