PDS4 Training Exercise

2017 FALL AGU MEETING

NEW ORLEANS, LA

DECEMBER 12-15, 2017

Objectives

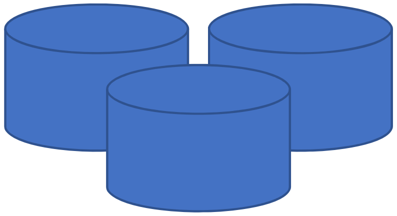
- What you will learn…
  - Key components of a PDS4 archive
  - Basic structure of PDS4 metadata
- What you will do…
  - Produce a set of valid PDS4 archive products

Training materials are available online:

https://pds.jpl.nasa.gov/pds4/training/2017-agu/index.shtml

What are PDS and PDS4?

The **Planetary Data System** (PDS) is NASA’s repository for the distribution and long term preservation of NASA planetary data.



The **PDS Archive** is the digital data repository maintained by PDS.

The **PDS Standard** are requirements and constraints designed to insure the usability of data in the PDS Archive throughout the lifetime of the archive.

PDS4 is the latest version of the PDS Standard.

PDS4 is **not** a data format!

Fundamental Data Structures

PDS4 supports science data files which are describable using one of the following fundamental data structures:

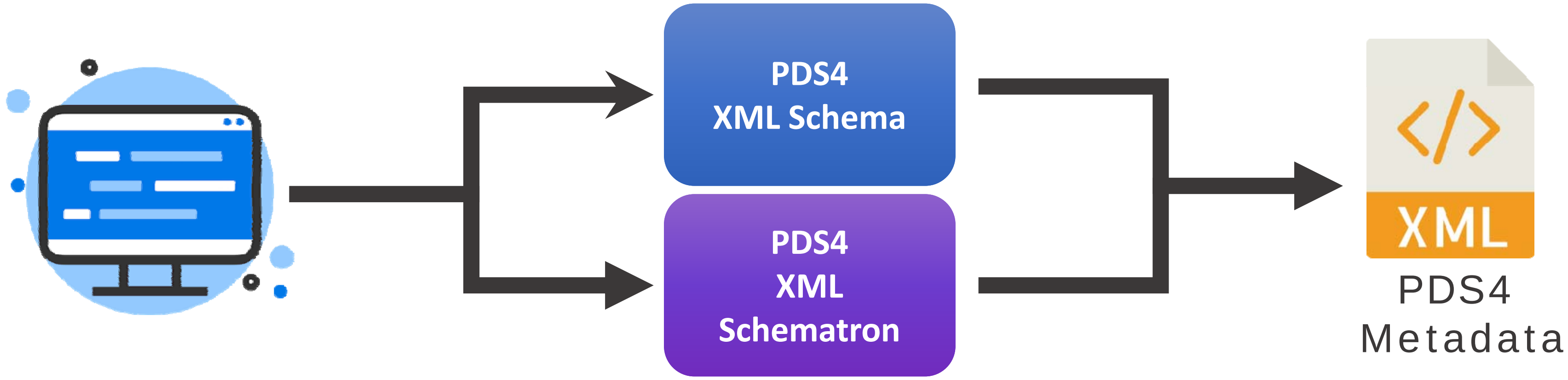
Array – homogenous binary structures of 1 to 16 dimensions in which all of the elements have the same data type.

Table – ASCII or binary data with a repeating record structure made up of fixed-width fields.

Parsable Byte Stream – ASCII data with a repeating record structure made up of variable width fields separated by a field delimiter (e.g. CSV).

Encoded Byte Stream – Data which is formatted according some well known standard (e.g. PDF).

PDS4 Implementation



PDS4
XML Schema

PDS4
XML
Schematron

PDS4
Metadata

The structure and content of PDS4 metadata is defined by a formal **Information Model**.

PDS4 is implemented in XML and expressed in terms of XML **Schema** and **Schematron** files.

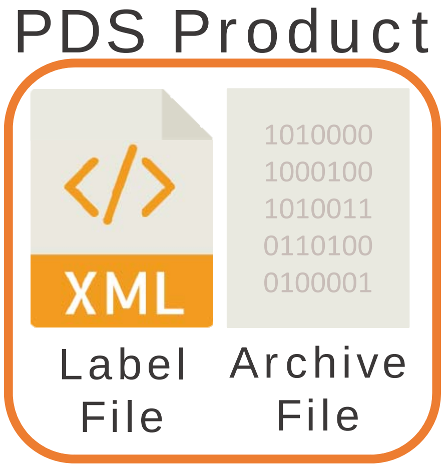
- Schema define the metadata structure
- Schematron provide rule-based constraints on elements and content

XML

PDS4 metadata are formatted according to the eXtensible Markup Language (**XML**) standard.

- XML is a markup language (similar to HTML).
- Values are placed between opening and closing tags:
  <tag>Value</tag>
- There are two types of tags:
  - Attributes – simple elements containing values between their tags
  - Classes – complex elements consisting of nested hierarchies of attributes and other classes
- XML is required to be “well-formed”:
  - A single root class
  - All elements consist of matching start and end tags
  - Start, end, and empty tags must be correctly nested
  - Characters are resticted to UTF-8 (use of “<”, “>”, and “&” is restricted)
  - Element order is prescribed

PDS4 Products



PDS Product

Label File

Archive File

- A file containing PDS4 metadata is called a **PDS Label.**
- A PDS label along with the file or files that it describes constitutes a **PDS Product**.
  - PDS4 labels are co-located with the files that they describe.
  - All elements of a PDS4 archive are products.

PDS4 Archive Organization

- There are 3 primary types of products in PDS4:
- **Basic Products** are the smallest unit of a PDS4 archive. They consist of an individual label and the associated file or files.
- Related basic products of the same type may be grouped together into a **Collection**.
- Related collections may be grouped together into a **Bundle**.

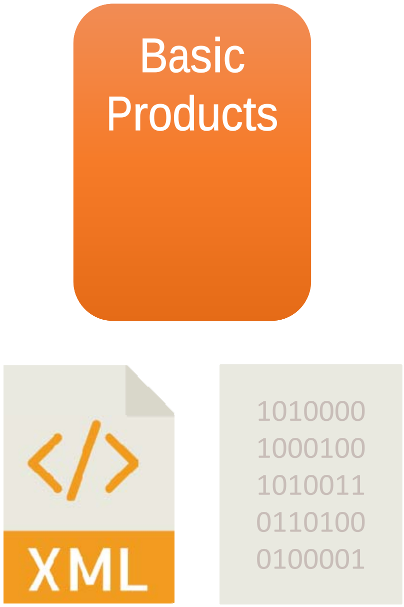
Basic Products

Collection Products

Bundle Products

Basic Products

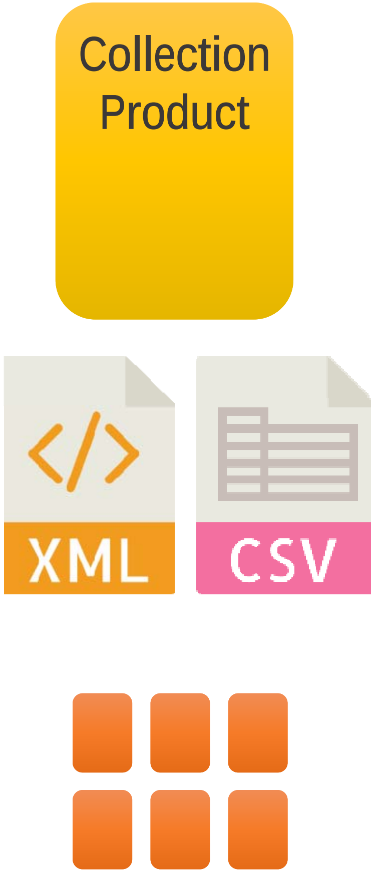
- Basic products consist of:
  - One or more archive files
  - A PDS label file describing the content, and as appropriate, the structure of the labeled file or files
- Basic products may be of many types (data, browse, documents, etc.).
- Basic products are frequently referred to generically as “products”.



Basic Products

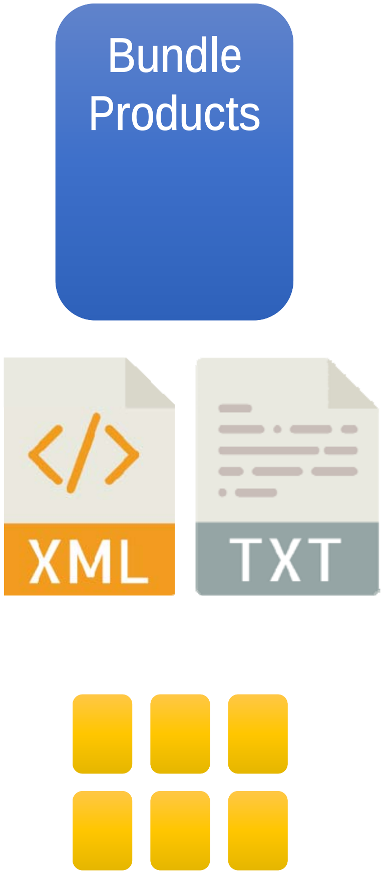
Collection Products

- Collection products consist of two files:
  - a **Collection Inventory**, a CSV file which contains a list of all of the basic products which are members of the collection
  - a PDS label file which contains a description of the basic products contained in the collection, as well as a description of the collection inventory file
    - The collection label file may optionally roll-up any metadata contained in the individual member products (e.g. targets, time ranges, etc.).



Bundle Products

- A bundle product consists primarily of a PDS label file.
  - A list of the collection products which are members of the bundle is included directly in the label.
  - Optionally the bundle label file may also roll-up any metadata contained in the individual member collections (e.g. targets, time ranges, etc.).
  - A bundle product may also include an optional "Readme" text file.
    - The Readme file must either be plain ASCII text or UTF-8 format.
    - It should include an overview of bundle content and organization.



Logical Identifiers (LIDs)

- LIDs must be unique across PDS
- LIDs take the form of a Uniform Resource Name (**URN**).
  - **Do not** specify physical location

urn:nasa:pds:*bundle*:*collection*:*product*

- Maximum length: 255 characters.
- Segments are delimited by colons.
- Allowed characters: **lower case** letters, digits, dash, period, and underscore.
- Segments:
  - 1: URN identifier (static)
  - 2-3: Archiving agency (static)
  - 4: Bundle identifier
  - 5: Collection identifier
  - 6: Product identifier
- Within the bundle, collection, and product identifiers dash, period, and/or underscore may be used as delimiters.

LID Segments

urn:nasa:pds:*bundle*:*collection*:*product*

- Segments 1-3 are static
- ***bundle*** is an identifier for the bundle and is included in the LID of all members of the bundle
  - Must be unique across all of PDS
- ***collection*** is a collection identifier and is included in the LID of all members of the collection
  - Must begin with the collection type (data, document, etc.)
  - Must be unique within the bundle
- ***product*** is an identifier for the individual product
  - Must be unique within the collection

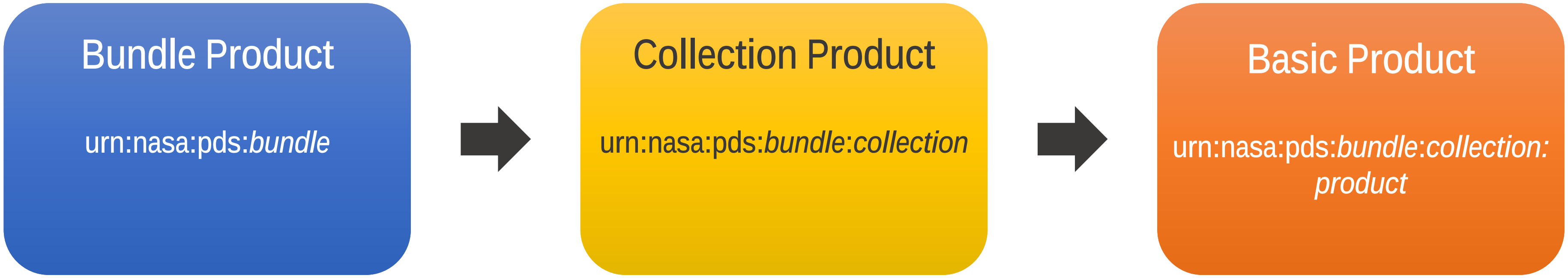
Version Identifier

urn:nasa:pds:*bundle*:*collection*:*product*::*ver*

- The product Version Identifier (VID) may be appended to the LID to form a **LIDVID**.
  - A double colon (::) is the delimiter to separate the VID from the LID.
- Internal references may be given either as **LIDs** or **LIDVIDs**.
  - A LID refers to a product without specifying a specific version.
  - A LIDVID refers unambiguously to a specific version of the referenced product.
- PDS4 products primarily use a 2 component VID: *M.n*
  - The major component (*M*) starts from “1”.
  - The minor component (*n*) starts from “0”; resets whenever *M* is incremented.

Archive Bundle LID

- The bundle product defines the bundle portion of the LID for its member collections.
- The collection product defines the collection portion of the LID for its member basic products



Bundle Product

urn:nasa:pds:*bundle*

Collection Product

urn:nasa:pds:*bundle:collection*

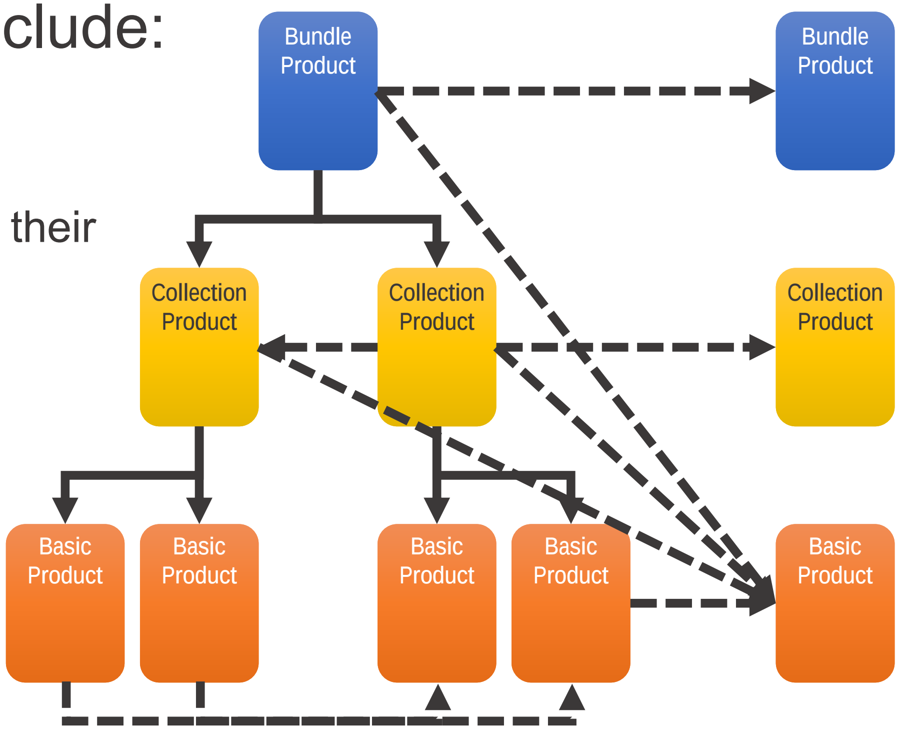
Basic Product

urn:nasa:pds:*bundle:collection: product*

LID/LIDVID Usage

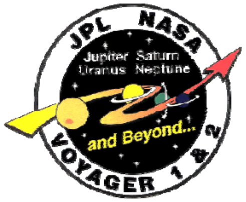
LIDs and LIDVIDs are used to identify a relationship between two PDS4 labeled products. Examples include:

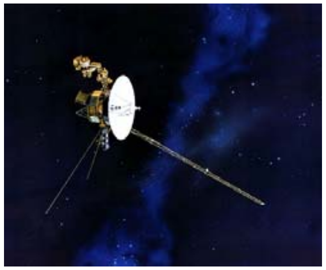
- Inventories point from a product to its member products (solid lines):
  - Bundle and collection inventories point to their respective collection and basic member products
- Internal References indicate a direct relationship between products (dashed lines):
  - Bundle, collection, and basic products referencing related products of the same type
  - Bundle, collection, and basic products to relevant basic products (e.g. documents, context products, etc.)

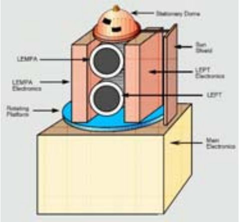


Context Products

- Describe physical or conceptual objects which are not physically part of the PDS archive (e.g. institutions, missions, spacecraft, instruments, targets, etc.)



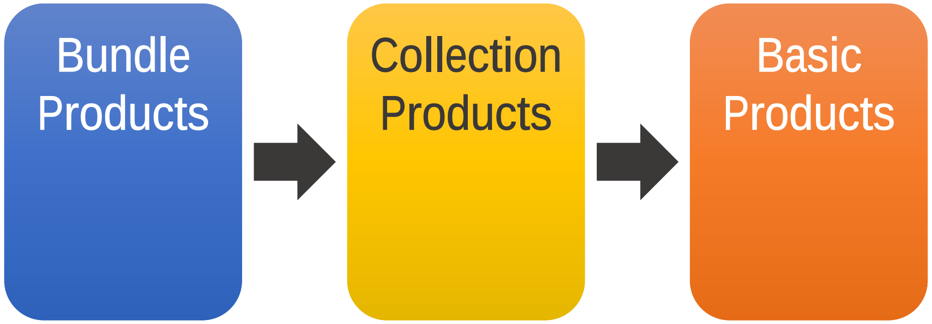




- Provide the ability to indicate the association or provenance of other types of archive products by referencing the context product
- Under the stewardship of (managed and maintained by) the PDS Engineering Node
- Not designed to be user documentation for those objects

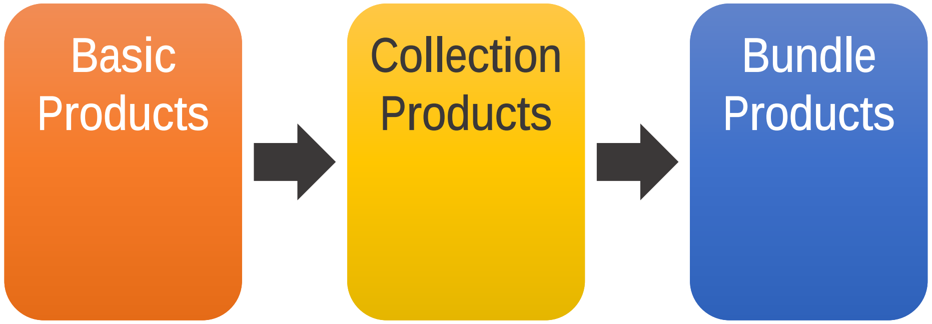
Archive Generation Procedure

Product planning and design should go from top down:



- Collections inherit the bundle ID from the LID of their parent bundle; basic products inherit the bundle and collections IDs from their parent bundle and collection.

Product generation should go from bottom up:



- Metadata from the basic products needs to be rolled up into the collection and bundles.

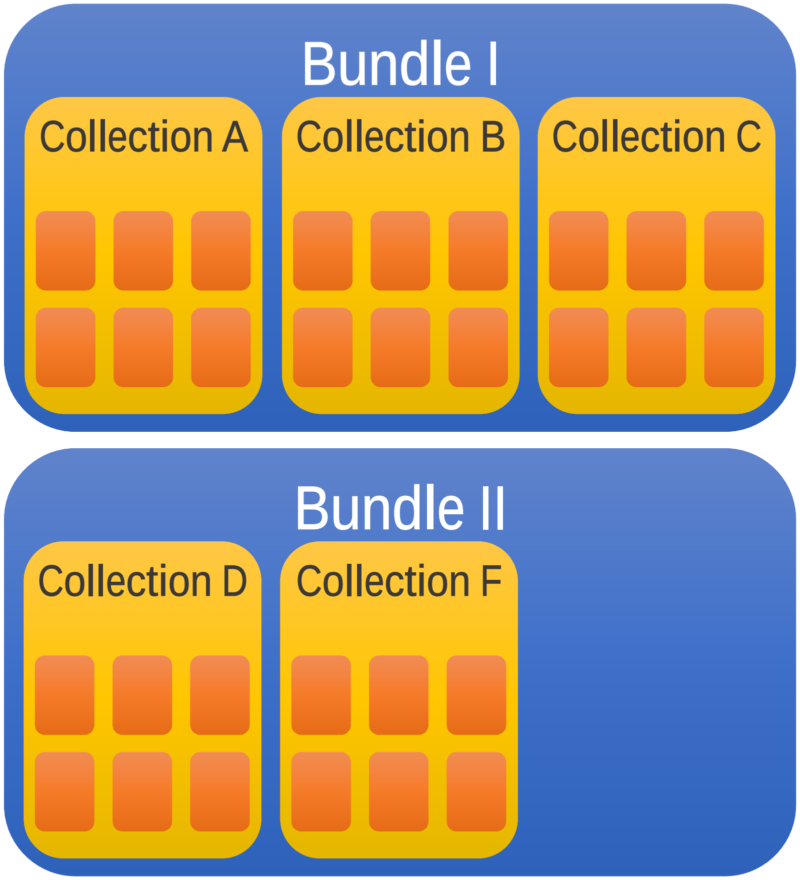
Archive Design

There is no hard and fast rule governing how a PDS4 archive is to be organized.

Data providers may want to consider the following questions:

- What organization makes sense for the data?
- What are other data providers on the project planning to do?
- What are data users likely to find the most useful?

Consult with your curating node!



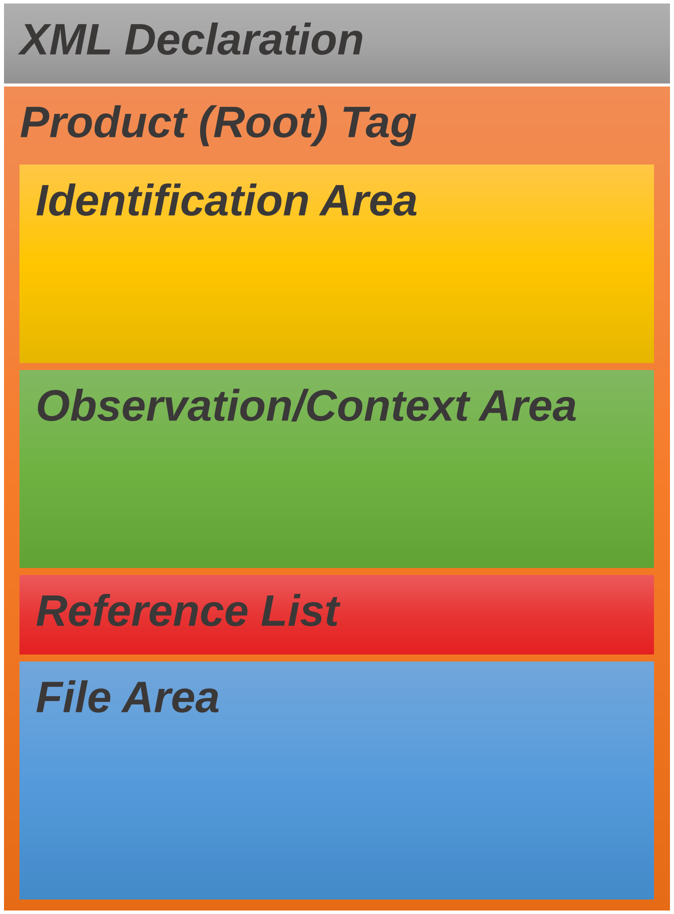
Anatomy of a PDS4 Label

XML Declaration

- XML identification tag
- Schematron location information

Product Tag

- Root product type tag
- Namespace declarations
- Schema location information



Anatomy of a PDS4 Label

Identification Area

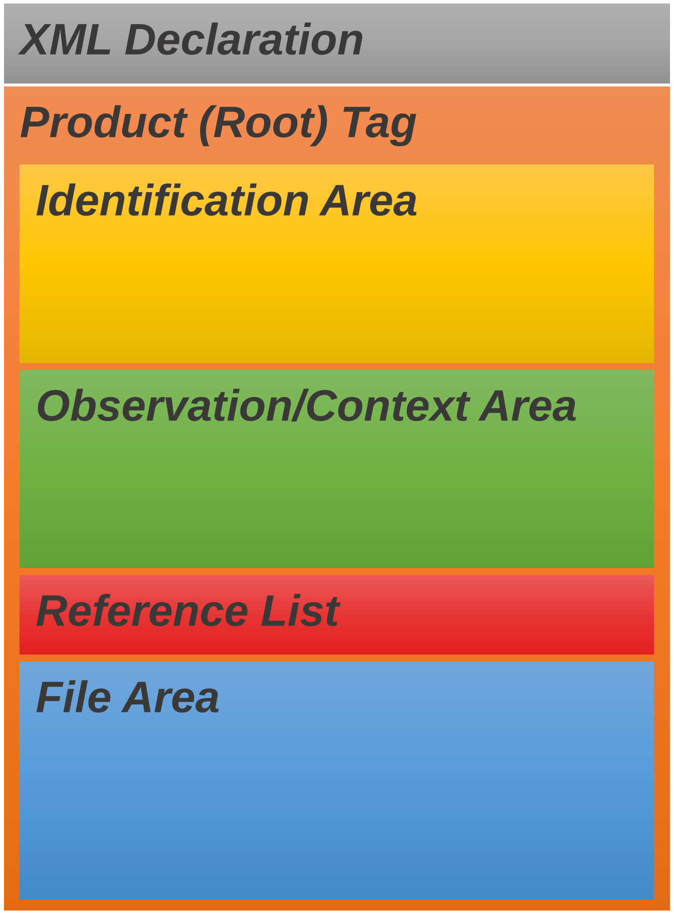
Contains product identifying information

- LID & VID definition
- Authorship/citation information (optional)
- Product modification history (optional)

Observation/Context Area

Contains product provenance/background

- Observation time
- Scientific content description (science discipline, data processing level, wavelength range, etc.)
- Target
- Source (mission, observatory, instrument, etc.)
- Discipline specific metadata (image display settings, geometry, etc.)
- Mission specific metadata



XML Declaration

Product (Root) Tag

Identification Area

Observation/Context Area

Reference List

File Area

Anatomy of a PDS4 Label

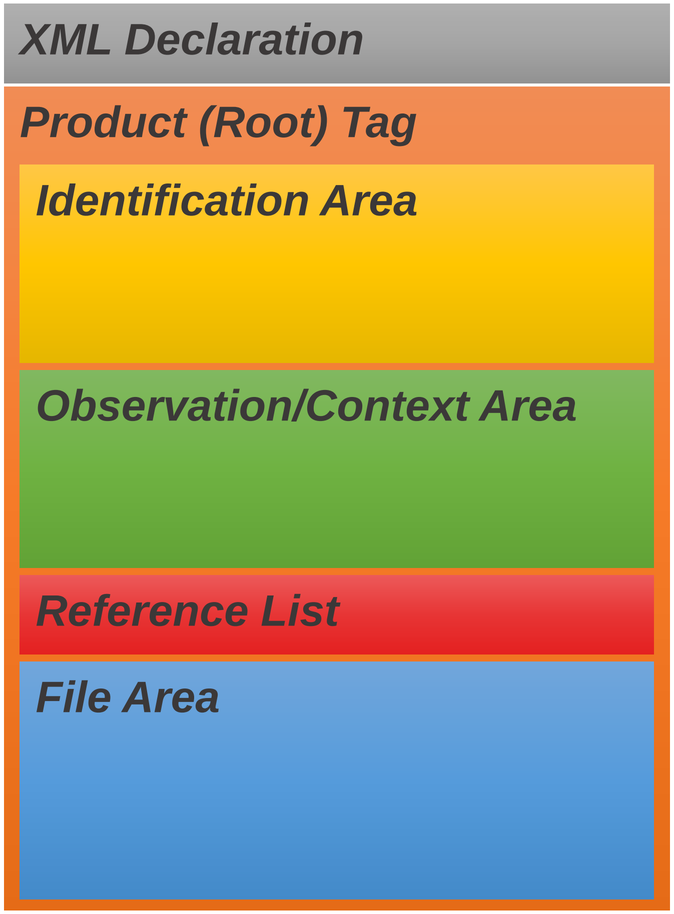
Reference List

Contains links to other PDS4 products (by LID/LIDVID) and external publications

File Area

Contains a description of the labeled file

- File name
- File statistical information (optional: size, creation date, MD5 checksum)
- File format information
- Data file structural information
  - Array element descriptions
  - Table record and field descriptions



PDS4 Build-A-Bundle Exercise

1) Archive Bundle Organization Design
2) Define Bundle and Collection Identifiers
   - Bundle and collection LID definition
   - Basic product LID formation rule
3) Generate Document and Document Collection Products
4) Generate Data and Data Collection Product Labels
5) Generate Bundle Readme and Label Files

Exercise Wrap-Up

Most Important Component

COMMUNICATION!

Make certain to identify the PDS Discipline Node that will be curating your archive early in the process and communicate with them regularly!

PDS Discipline Node Contacts

- Atmospheres Node

  Lynn Neakrase
  +1(575)646-1862
  lneakras@nmsu.edu
- Cartography and Imaging Sciences

  Lisa Gaddis
  lgaddis@usgs.gov
- Geosciences

  Ed Guinness
  guinness@wunder.wustl.edu
  Susie Slavney
  slavney@wunder.wustl.edu
- NAIF (SPICE)

  Boris Semenov
  boris.semenov@jpl.nasa.gov
- Planetary Plasma Interactions

  Joe Mafi
  jmafi@igpp.ucla.edu
- Ring-Moon Systems

  Mitch Gordon
  mgordon@seti.org
- Small Bodies

  Anne Raugh
  raugh@astro.umd.edu
  Jesse Stone
  jstone@psi.edu

Evaluation

Thank you for participating in our PDS4 training exercise. We would really appreciate your feedback on the quick survey below. Your answers are anonymous and are helpful to the development and improvement of our future training sessions.

https://goo.gl/forms/KmoOkTZrv1pgSs2z2

Thank you for your time!

Attributions

Backup

LID Bundle Identifier

urn:nasa:pds:*bundle*:*collection*:*product*

- Must be unique within PDS
- Bundle identifiers typically take the form:

mission-instrument[-description]

  - *mission* = The mission ID
  - *instrument* = The instrument ID
  - *description* = A description (optional) to help to distinguish the bundle from others from the same mission and instrument

- Examples: ladee_nms, maven-swea-calibrated

LID Collection Identifier

urn:nasa:pds:*bundle*:*collection*:*product*

- Must be unique within the bundle
- Starts with the collection_type value (lowercase)
- Collection identifiers typically take the form:

collection_type[-description]

  - *collection_type* = collection_type value (i.e. data, document, etc.)
  - *description* = A description (optional) to help to distinguish the collection from others of the same type within the bundle (e.g. data type, mission phase, etc.)

- Examples: data, data_calibrated, data-svy-3d

LID Product Identifier

urn:nasa:pds:*bundle*:*collection*:*product*

- Must be unique within the collection
- Typically consists of the base file name of the labeled file
- Examples:

  nms_cal_hk__36127_20131203_104228,

  mvn_swe_l2_svy3d_20161208
- Design notes:
  - Uppercase characters must be converted to lowercase.
  - File version numbers, and other variable portions of the file name should be omitted from the product identifier.