LIDs, VIDs, and LIDVIDs

To keep the PDS data archive organized, every product is assigned a unique identifier. This Logical Identifier (LID) is not only unique to your archive but is unique through all PDS.

For any questions about creating and assigning LIDs contact your PDS node consultant.

LID Format

urn:nasa:pds:<bundle_id>:<collection_id>:<product_id>

Rules

LIDs are constructed by concatenating fields of characters. The fields are separated by colons.

- LIDs are restricted to lower-case letters, digits, the dash, the period, and the underscore.
- Bundle LIDs are constructed by appending a unique bundle-specific identifier to the agency identifier, such as urn:nasa:pds.

  Bundle LID = urn:nasa:pds:<bundle_id>

  Since all PDS bundle LIDs are constructed this way, the bundle LID will be globally unique.
- Collection LIDs are constructed by appending a unique collection identifier to the parent bundle's LID, for example:

  Collection LID = urn:nasa:pds:<bundle_id>:<collection_id>

  Since all PDS collection LIDs are constructed in this way and the collection identifier is unique within the bundle, the collection LID will be globally unique.
- Basic Product LIDs are constructed by appending a unique product identifier to the parent collection's LID.

  Product LID = urn:nasa:pds:<bundle_id>:<collection_id>:<product_id>

  Since the product LID is based on the collection LID, which is unique across PDS, the product LID will be globally unique.

Where to Start

Knowing how you want to organize your PDS4 archive is the first step. Every LID will start with:

urn:nasa:pds

For the next step, you need to determine what you will name your bundles.

Bundle Organization

Bundles are the highest level of organization in a PDS4 archive. They can be organized in a wide variety of ways. Some suggestions include:

- Documents, Raw Data, Derived Data, etc.…
- Mission documents, Instrument 1, Instrument 2, etc.…
- Orbit 1, Orbit 2, etc.…

There are many ways to break your archive into related groups of products and you can make as many bundles as you need.

For example, if your study is called GECKO and you choose to separate your archive products into Documents, Raw Data, and Derived Data, then your potential bundle LIDs could be:

urn:nasa:pds:gecko_documents

urn:nasa:pds:gecko_raw_data

urn:nasa:pds:gecko_derived_data

The most important part of creating your bundle LIDs is that they are unique to all other bundles within the PDS. Confirm with your Node representative that your bundle LIDs are unique. Any additional identifiers such as the collection_id and product_id do not have to be unique. Their uniqueness within the PDS will come from the bundle_id.

Collections

Collections are the next level of organization below bundle. As with organizing your study into bundles, there are many ways you can organize your bundle into collections.

Continuing with the GECKO example, let's look at possible ways to organize the Raw Data bundle into collections:

- Observation days
- Target names
- Instrument names
- Orbits

Some example collection LIDs could be:

urn:nasa:pds:gecko_raw_data:target1

urn:nasa:pds:gecko_raw_data:camera1

The gecko_documents bundle could be separated into: Study documents, Instrument information, Target information, Host information, etc.… Some example collection LIDS could be:

urn:nasa:pds:gecko_documents:mission

urn:nasa:pds:gecko_documents:target1

Basic Products

Basic Products are the actual files you are storing in the archive. These are grouped into collections and therefore their LID will include the bundle_id and collection_id of the collection in which they are included.

For our GECKO study, the basic products within the *urn:nasa:pds:gecko_documents:target1* collection could include: target1 information, observation summaries, weather reports, etc.… Some example collection LIDS are:

urn:nasa:pds:gecko_documents:target1:day1_obs_summary

urn:nasa:pds:gecko_documents:target1:day1_cloud_coverage_report

urn:nasa:pds:gecko_documents:target1:target_description

VID Format

urn:nasa:pds:<bundle_id>:<collection_id>:<product_id>::<version_id>

The way PDS4 keeps track of archival versions is with a version identifier (VID). The way we denote the VID is through two integers separated by a period. The integer before the period is incremented when there are major changes to the product and the integer after the period is incremented for minor changes. If there is any confusion on what constitutes a major or minor change to a product, contact your Node representative.

The VID is used in a variety of ways. It is listed in the Identification Area of a label and in the Modification History Area

Rules

Version IDs are used for all types of products, including basic products, collections, and bundles.

- VIDs are appended to LIDs by a double colon ("::").
- VIDs must be of the form M.n, where M denotes a major version and n denotes a minor version.
- The major number (M) is initialized to 1 for archive products. Whenever the major number (M) is incremented, the minor number (n) is reset to 0.
- Neither M nor n should be prepended with zeros; each is simply incremented as an integer. Thus 1.1 and 1.10 are different versions, and 1.01 is invalid.

LIDVID Format

urn:nasa:pds:<bundle_id>:<collection_id>:<product_id>::<version_id>::M.n

PDS4 connects the Logical Identifier (LID) to the Version Identifier (VID) with a double colon. This combination is called a LIDVID. All products included in an archive (bundles, collections, and basic products) have a LIDVID.

Bundles

urn:nasa:pds:gecko_documents::1.5

Collections

urn:nasa:pds:gecko_raw_data:target1::2.0

Basic Products

urn:nasa:pds:gecko_documents:target1:day1_obs_summary::1.0

If a bundle or collection is updated by adding new products or making changes to its products, you must increment the VID. Only increment the VID of individual basic products within a bundle or collection when those specific products are updated. Each product's VID is initially 1.0.

For more information and examples, see chapter 5 of the Data Providers Handbook.