# [CCB-204] Define and enforce best practices for discipline and project dictionaries.

Created: 20/Dec/17 Updated: 14/Nov/19

| | |
|---|---|
| **Status:** | Open |
| **Project:** | PDS4 Standards Change Control Board |
| **Component/s:** | External Agencies / Interfaces, Information Model, PDS Tools |
| **Affects Version/s:** | 1.E.0.0 |
| **Fix Version/s:** | None |

| | | | |
|---|---|---|---|
| **Type:** | Enhancement / Improvement | **Priority:** | Normal / Non-urgent |
| **Reporter:** | Anne Raugh | **Assignee:** | Emily Law |
| **Resolution:** | Unresolved | | |
| **Labels:** | Open | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original Estimate:** | Not Specified | | |

| | |
|---|---|
| **Problem Statement:** | Define and enforce best practices for discipline and project dictionaries. |
| **target date:** | 01/Mar/18 |
| **Proposed Solution:** | Document the constraints/best practices.<br>Upgrade LDDTool to check and report violations as possible and appropriate.<br>Add an attribute to the Ingest_LDD structure to indicate whether the target namespace is for a PDS discipline dictionary or a mission/project dictionary; modify LDDTool to modify its behavior based on this attribute rather than the current command line option.<br>Remove from LDDTool the current "-M" command option. |
| **Dependencies / Contingencies:** | This is part of a general upgrade of LDDTool and standardization of local dictionary procedures and requirements. |
| **Impact Statement:** | Impact:<br>-- PDS Tools -- The requested change has moderate impact on the LDDTool software.<br><br>No Impact:<br>-- Information Model<br>-- Standards Reference |

| | |
|---|---|
| | -- Concepts Document<br>-- APG<br>-- DPH<br>-- External Agencies<br>-- ISO Standards<br>-- PDS Website<br>-- PAG |
| **Technical Assessment:** | The request is to improve the validation of Ingest_LDD files. This change adds constraints on attributes in local dictionaries (both discipline and mission/project). These constraints will be enforced by enhancing LDDTool to check for violations and report them as warnings or fatal errors. This change request is reasonable. |
| **System Impact:** | backwards compatible |

**Description**

There are some conventions for creating local dictionaries that discipline dictionary stewards should either follow or avoid in order to remain consistent with the core pds: dictionary style and conventions. These conventions should also be recommended to project dictionary stewards as best practices, but projects may choose not to conform. LDDTool should be able to determine whether a discipline or project dictionary is being built, and report lack of adherence to these conventions as either an error (for discipline dictionaries) or a warning (for project dictionaries), where possible

The proposed conventions/best practices that can be enforced by LDDTool are:

1. Local dictionary attributes named "type" or with names ending in "_type" must have permissible value lists. In the pds: and discipline namespaces these attributes are associated with search facets as well as value validation.
2. No attribute should be named "unit", "units", "unit_of_measure", or similar, or end in any of those strings. The pds: namespace uses XML attributes and unit classes to define units to enable searching across data from different sources on numeric fields with units of measure. This also ensures that the unit of measure is directly associated with the attribute value to which it applies.

These conventions/best practices might be enforceable by LDDTool (technical assessment needed to determine this):

1. No class defined with <element_flag> set to "true" may be a component at any level of another class with <element_flag> set to "true". In other words, if Class_A and Class_B both <element_flag> set to "true", then Class_A cannot include Class_B or any class that contains Class_B, and conversely. This effectively requires that dictionaries define unique "entry points" - specific, high-level classes that act as containers for the dictionary content.
2. An attribute that is defined as "nillable" must be a required attribute of at least one class. This follows the PDS core namespace design principle that attributes are either optional (and omitted when they do not have a value), or required but nillable (with a required "nilReason" attribute to explain why the value is not present).

In addition, the following constraint/best practice likely can not be enforced by LDDTool, but should be documented with a recommended enforcement

method (like "provide a test label to demonstrate implementation", for example):

1. When a local dictionary uses a pds:Local_Identifier_Reference class to identify another class in the same label, it must also include a Schematron rule to validate that the local_identifier referenced is associated with a class of the appropriate type. For example, if a local dictionary class is intended to describe an Array_2D_Image, it should contain a pds:Local_Internal_Reference class to establish the link to the specific Array_2D_Image it is describing, and there must be a Schematron rule that validates that the pds:local_identifier_reference value corresponds to a pds:local_identifier of an Array_2D_Image object (and not a Table_Character, for example).

**Comments**

Comment by Edward Guinness [ 14/Nov/19 ]

No class defined with <element_flag> set to "true" may be a component at any level of another class with <element_flag> set to "true". In other words, if Class_A and Class_B both <element_flag> set to "true", then Class_A cannot include Class_B or any class that contains Class_B, and conversely. This effectively requires that dictionaries define unique "entry points" - specific, high-level classes that act as containers for the dictionary content.

Question: Does the above requirement mean that classes within a heirarchy where the top-level class has element_flag = true cannot also have a value of element_flag = true.

Generated at Tue Dec 03 19:51:29 PST 2019 by Steven Hughes using Jira 8.4.1#804002-sha1:94e96d6294939fc41f873484cf96b18ab6f2be0a.