LDDTool

Last Published: 2015-09-09 | Version: 0.2.0.0

Software Documentation

- **Operation**

Download

- Data Design Working Group -> Resources -> LDDTool_0.2.0.0_VID.zip (Binary Package)

Operation

This document describes how to operate the LDDTool. The following topics can be found in this document:

Note: The command-line examples in this section have been broken into multiple lines for readability. The commands should be reassembled into a single line prior to execution.

Tool Execution

The LDDTool can be executed in various ways. This section describes how to run the tool, as well as its behaviors and caveats.

Command-Line Options

The following table describes the command-line options available:

Command-Line Option	Optional/ Required	Description
-p, --PDS4	Required	Set the context to PDS4.
-l, --LDD	Required	Process a local data dictionary input file.
-c, --Class	Optional	Write definitions for class elements.
-a, --Attribute	Optional	Write definitions for attribute elements
-J, --JASON	Optional	Write the master data dictionary to a JASON formatted file.
-m, --merge	Optional	Generate file used to merge the local dictionary into the

		master dictionary
-M, --Mission	Optional	Generate files with Mission included in the targetNamespace
-n, --nuance	Optional	Write nuance property maps to LDD schema annotation in JASON
-s , --sync	Optional	Use local namespace + information model version as output file names.
-1 --IM Spec	Optional	Write the Information Model Specification with LDD.
-h, --help	Optional	Display LDDTool usage.
filename	Optional x n	Input file name of a referenced LDD.
filename	Required	Input file name of this LDD (with or without a directory path; with or without the file extension '.xml')

Running the LDDTool

The LDDTool requires the passing in of an XML document that locally defines new classes and attributes to be used in PDS4 product labels. The XML document must conform to the PDS4 Ingest_LDD schema. The file specification allows the input file to either be local to the LDDTool directory or remote. The name of the input file is used to create the names of the output files. For information about creating the XML document please see the "Creating an LDD" document.

The contents of the document are parsed, validated, and used to create at least four output files. The XML schema and schematron files contain the class and attribute definitions and may be included into a PDS4 product label schema in either the discipline or mission areas. The process report file lists the defined classes and attributes and provides validation information. A log file is also written to the standard "system out" stream and provides more general processing and error information. Optionally a database instance file can be created that is used to merge the LDD information into the master PDS4 database.

LDDTool can be executed from the command-line as follows: **Command-line**	**Action**
lddtool -h	• Display LDDTool usage
lddtool -pl inputfile	• Process inputfile.xml, • Validate inputfile and display the status of the validation, • Generate report files
lddtool -plm inputfile	• Process inputfile.xml, • Validate inputfile and display the status of the validation, • Generate file used to merge the local dictionary into the

	• master dictionary • Generate report files
lddtool -plc inputfile	• Process inputfile.xml, • Validate inputfile and display the status of the validation, • Write definitions for class elements • Generate report files •
lddtool -plm inputfile > log.txt	• Process inputfile.xml, • Validate inputfile, • Generate file used to merge the local dictionary into the master dictionary • Generate report files • Write the status of the validation to a log file

Run-time Configuration

LDDTool is executed from the 'bin' directory. The 'bin' directory is where the inputfile.xml file can be located and where the output, report, and log files will be written.

Input File

The input file to LDDTool is an XML document that contains a semantically and syntactically valid local dictionary template. The input file must conform to the PDS4 Ingest_LDDTool2 schema.

Notes:
1. See Section 7.1 of the Data Providers Handbook (DPH) for a description of steps to create an XML label template from the master-schema.
2. See Section 7.2 of the DPH for a description of how to populate a label template.

Output, Report, and Log Files

The output and report files are named in accordance with the name of input file (e.g., input_123.xml will generate output / report files having a name of input_123 and appropriate file extensions).

File Extension	Type of File
.xsd	XML schema file
.sch	Schematron file
.csv	Data Dictionary information in csv format
.txt	Report of processing / status in text format
.pont	Ontology file for merging local dictionary into master dictionary

Report Formats

This section describes the contents of the LDDTool report formats. The tool has different reporting formats depending on the behavior of the tool as defined by the command-line options.

XML Schema file (xsd)

If LDDTool detects a compliant XML document as input, the tool will generate a compliant XML-schema (XSD) file. This is the local-dictionary schema that is referenced in your products labels. See Section 11 of the Data Providers Handbook (DPH) for more information on how to reference a local-dictionary schema (XSD) in a PDS4 product label.

Schematron file (sch)

If LDDTool detects a compliant XML document as input, the tool will generate a compliant XML-schematron (XSD) file. This is the local-dictionary schematron file that is referenced in your product labels --- assuming your local dictionary additionally requires a schematron file. The schematron file will contain references to 'validation' rules if any were specified in the input file.

Report files (csv, txt)

LDDTool generates a number of report files that list the status of the validation / ingest steps. Inspection of these files is a good thing to ascertain if any errors / warnings were detected.

Ontology file (pont)

If LDDTool detects a compliant XML document as input, the tool will generate a conformant ontology file. This file is used to 'migrate' the locally defined attributes and classes into the master-schema. This file is to be sent to the curator of the master-schema.

Processing Status

If directed to generate a log file via the command-line, LDDTool will generate a log file that contains the processing status of the validation request. If not directed to generate a log file, LDDTool will display the processing status of the validation request. The following is an example of a successful validation request:

```
>>info - LDDTOOL Version: 0.1.4
>>info - Input File Name: IngestLDDTool_2
>>info - Date: Mon Oct 29 11:04:16 PDT 2012
>>info - JAVAHOME: C:/Program Files/Java/jre6
>>info - PARENT_DIR: E:/LDDTool/LDDTool/bin/..
>>info - SCRIPT_DIR: E:/LDDTool/LDDTool/bin/
>>info - LIB_DIR: E:/LDDToo/LDDTool/bin/../lib
>>info - USERNAME: rsmith
>>info - Found input file: IngestLDDTool_2.xml
>>info - Found required file: E:/LDDTool/LDDTool/bin/../Data/MDPTNConfigClassDisp.xml
>>info - Found required file: E:/LDDTool/LDDTool/bin/../Data/DMDocument.pins
>>info - Found required file: E:/LDDTool/LDDTool/bin/../Data/glossary.pins
>>info - Found required file: E:/LDDTool/LDDTool/bin/../Data/UpperModel.pont
>>info - Found required file: E:/LDDTool/LDDTool/bin/../Data/dd11179.pins
>>info - LDDTOOL Exit
```

Should the results of the validation request not reflect the above, something terrible has gone wrong and you might consider contacting the curator of LDDTool for further instructions / support