

CONVERTING DATA FORMATS FROM PDS3 TO PDS4 WITH THE PDS GENERATE TOOL. P. E. Geissler¹ U.S. Geological Survey, 2255 N. Gemini Drive, Flagstaff, AZ, USA (pgeissler@usgs.gov).

Introduction: Since 2011, NASA has required that new data submitted to the Planetary Data System (PDS) for inclusion in their archive be in the modern PDS4 format [1-3]. This has necessitated both migration of data formatted in the older PDS3 format and establishment of new methods and tools for creating data archives that are fully compliant with PDS4 formats [4-6]. The PDS provides a number of tools to support users in creating and working with archived data in PDS4 formats (see <https://pds.nasa.gov/tools/about/>). A first-order need is to convert data formats from PDS3 to PDS4; the PDS Generate Tool supports this task (<https://pds.nasa.gov/tools/about/generate/>).

Here we provide a step-by-step guide to automatically generating PDS4 labels from a valid PDS3 archive for users working on a linux operating system. First, you must create a valid PDS4 prototype label in XML with all the information and metadata you wish to preserve for a data product. These are described in the PDS Data Providers Handbook [4], and examples of valid PDS4 data organization, products and labels are provided (see <https://pds.nasa.gov/datastandards/documents/examples/>). Second, you must convert the prototype label to a label template by replacing the specific information with Apache Velocity (<https://velocity.apache.org/>) variables or expressions. Finally, you must run the Generate Tool together with the label template for batch conversion from PDS3 to PDS4. These three steps are detailed below.

Building a Prototype PDS4 Label: To begin with, you need a valid PDS4 label that contains all the information that you wish to preserve from the original PDS3 label. A

good place to start is with the PDS Transform Tool (linux commands are shown in green):

```
setenv JAVA_HOME /usr/
transform-1.6.0/bin/transform
MYPRODUCT.LBL -f pds4-label
```

This will create a valid PDS4 label that is not quite complete, so you will need to add any missing information manually using example PDS4 labels for similar products. Don't forget to reference all the schematrons (rules and definitions) used in your label to be certain that your additions are valid. For example, if you add map projection information you must reference PDS4_CART_1900.sch and PDS4_CART_1900.xsd at the start of your PDS4 prototype label.

Make sure that your label is valid, using the PDS Validate Tool or the Oxygen XML Editor (with appropriate references to the schematron, etc.), and make sure that your product displays correctly in the PDS4 Viewer (http://sbndev.astro.umd.edu/wiki/PDS4_Viewer). Then you are ready for the next step.

Turning Your Prototype Label Into a Label Template: Next, you must replace the specific information in your label with Apache Velocity variables and expressions to create a template for the rest of your PDS3 products. For example, if your prototype label has the following line:

```
<start_date_time>2009-11-
03T06:31:57.934Z</start_date_time>
```

you must replace it with:

```
<start_date_time>$label.START_TIME</start
_date_time>
```

You can use Apache Velocity variables and expressions to fill out the text of your label, for example:

```
#set($suffix=".IMG")
```

```
<local_identifier_reference>
  $label.PRODUCT_ID$suffix</local_identifier
_reference>
```

uses the variable `$suffix` to append “.IMG” to the `PRODUCT_ID`. You can also use Apache Velocity expressions to calculate numbers needed by your label, such as:

```
<cart:pixel_resolution_x
unit="m/pixel">$math.mul($label.UNCOMPRESS
ED_FILE.IMAGE_MAP_PROJECTION.MAP_SC
ALE , 1000.0)</cart:pixel_resolution_x>
```

to convert from km/pixel to m/pixel.

Test your template by using it to recreate your original label:

```
generate-0.15.0/bin/generate          -p
MYPRODUCT.LBL -t mytemplate.vm
```

Using the Generate Tool for Batch Conversion: When your template is complete, you can use it to convert the PDS3 labels for all similar products to PDS4. Below is an elementary python script for automatically converting PDS3 labels for products named `BIBQ*LBL`:

```
#!/usr/bin/env python
import os, fnmatch, shutil, time, hashlib, datetime
directory =
"/scratch/pgeissler/test_Cassini_RADAR_PDS4/"
template = "BIBQ_template.vm"
filePattern = "BIBQ*LBL"
```

```
for path, dirs, files in
os.walk(os.path.abspath(directory)):
    for filename in fnmatch.filter(files, filePat-
tern):
        filepath = os.path.join(path, filename)
        filename_root = filename.split(".") [0]
        labelname = filename_root.lower()+".xml"
        pds3labelname = filename_root+".LBL"
        labelpath = os.path.join(path, labelname)
        print labelpath
        print (filepath)
        print filename
command = "generate-0.15.0/bin/generate -p
"+filepath+" -t "+template
        print command
        os.system(command)
```

Conclusion: Most of the effort in converting from PDS3 to PDS4 is in building a pro-

totype PDS4 label. Fortunately, many examples of valid PDS4 labels are now available; if you don’t find an appropriate one, consult the appropriate PDS node. Use of the PDS Generate Tool simplifies the process downstream.

References: [1] Crichton et al. (2011) EPSC Abstracts, 6, #1733. [2] Beebe et al. (2010) AAS-DPS meeting #42, id.37.02; Bulletin of the American Astronomical Society, Vol. 42, p.967. [3] Hughes et al. (2018) Planetary & Space Science 150, pp. 43-49. [4] The PDS4 Data Provider’s Handbook, https://pds.nasa.gov/datastandards/documents/dph/current/PDS4_DataProvidersHandbook_1.11.0.pdf. [5] Planetary Data System Standards Reference, 1.9.0 (<https://pds.nasa.gov/pds4/doc/sr/current/>). [6] Gaddis and Hare, this meeting.