

Guide to Archiving TIFF/GeoTIFF Files in PDS4 as Observational Products

revision 8, July 2020

Trent Hare, thare@usgs.gov,

Table of Contents

Overview for TIFF/GeoTIFF	1
TIFF File Structure.....	2
PDS4 Archive Requirements.....	3
Convert to a PDS4-compliant TIFF/GeoTIFF using GDAL.....	3
Convert to a PDS4-compliant TIFF using LibTIFF tools	4
Methods to Test for TIFF Compliance for a PDS4 Archive	5
References	6

Overview for TIFF/GeoTIFF

This whitepaper introduces the restricted form of TIFF/GeoTIFF format as an accepted Planetary Data System (PDS v4) science data product or “observational” image format (see p. 12 of the Data Providers Handbook [1]). This builds on the precedent set by PDS4-compliant restricted forms of formats such as FITS and CDF. TIFF is a flexible file format for handling images with an associated descriptive header within a single file. TIFF supports the use of simple raw binary streams (e.g. Array_2D* or Array_3D*) for image data allowing the format to be fully compliant with PDS4 requirements. While TIFF supports several compression methods and image tiling, these options are not allowed within a PDS4 archive. TIFF/GeoTIFF helps the planetary community and PDS to realize both a compliant archival format and an immediately “usable” format supported by hundreds of graphics-based and scientific applications (including popular applications like Photoshop, MATLAB, ENVI/IDL, multitudes of GIS applications, and image libraries). Currently, many instrument teams and PDS Nodes already provide image data as GeoTIFF within “Extras” directories as part of their PDS3 archives, largely to enhance usability of their products.

The GeoTIFF format was originally created in the late 1990s by Dr. Niles Ritter while he was working at the NASA Jet Propulsion Laboratory and it was designed specifically to be used for geospatial image data. GeoTIFF is a public domain standard which allows geospatial information to be embedded within the Tagged Image File Format (TIFF). Such information or metadata allows the user to establish the spatial reference within the file, including, for example, the map projection, coordinate system, body size definition, and other such parameters.

Since almost all mapping and image processing software support GeoTIFF, it has emerged as a standard image file format worldwide. So much so that, as of February 2017, GeoTIFF was recognized by the Library of Congress as the preferred archival format for cartographic raster images. Also, in September 2019, GeoTIFF version 1.1 was adopted as an Open Geospatial Consortium (OGC) Standard [4]. Under this GeoTIFF version 1.1, the OGC also formalized the existing GeoTIFF 1.0 specification by integrating it into OGC’s standardization process. Lastly, the TIFF format has been defined in several International Organization for Standardization (ISO) standards (e.g. ISO 12234-2 [5], ISO 12639 [6]). The GeoTIFF format is fully compliant with TIFF 6.0, so software incapable of reading and interpreting the specialized geospatial metadata can still open a GeoTIFF file. Thus, it is understood that accepting the GeoTIFF version 1.1 format fundamentally means allowing for the TIFF 6.0 file format.

TIFF File Structure

The first version of the TIFF format was released in 1986 by Aldus Corporation for use in desktop publishing. Aldus published TIFF version 6 in 1992, which was subsequently updated with an Adobe Systems copyright after they acquired Aldus in 1994. Although Adobe owns the copyright, there is no need for a license to implement the software for reading and writing the TIFF format. A widely used TIFF library called *libtiff* was written and initially released, in 1988 by Sam Leffler while he was working for Silicon Graphics. All *libtiff* source code is either in the public domain or under a permissive BSD-like license and it is free to use for any purpose (including in commercial software). *Libtiff* is now chiefly maintained by the Open Source Geospatial Foundation (OSGeo) and it is freely available [7].

The TIFF format has a three-level hierarchy including a (1) Image File Header (IFH), (2) one or more directories called Image File Directories (IFDs) containing codes and their data, or a pointer to the data, and (3) the image data (see Figure 1). The TIFF file always begins with an 8-byte header, which gives basic information about the file such as byte order (Little Endian or Big Endian), TIFF file ID or Version Number (which should always be 42) and a pointer to the first IFD. Traversing the IFD(s) allows one to locate the pointer (offset bytes) to the image data. Each IFD contains one or more data structures called tags. Each tag is a 12-byte record that contains a specific piece of information. The TIFF file format uses 32-bit offsets which limits the output file size to 4 gigabytes. A simple variant of the original TIFF, called BigTIFF, which uses 64-bit offsets, allows for much larger file sizes.

To support a geospatial (Geo/GIS) extension within the GeoTIFF label, an extra set of IFDs have been defined and well standardized. The GeoTIFF format fully complies with the TIFF 6.0 specifications, and its extensions do not in any way contradict the TIFF standard, nor do they limit the scope of raster data supported by TIFF. Note that the TIFF header (the IFH and IFD) is not considered a PDS4 archival object. Thus, any critical information in the TIFF heading, including any GeoTIFF tags, must be translated into the PDS4 label including the offset bytes to the actual image data.

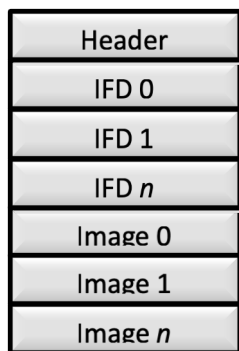


Figure 1. Recommended physical arrangement for a TIFF file. TIFFs must have an 8-byte header (IFH) and at least one Image File Directory (IFD). IFDs allows additional information which are not specified in the 8-byte header including the offset bytes to the image data. The offset bytes is the critical value required for any PDS4 label to allow a PDS4 reader to skip over the TIFF header (and any IFDs). The Image sections will be referenced in PDS4 labels as Array_2D* or Array_3D*.

PDS4 Archive Requirements

The PDS prefers clear, non-proprietary formats when archiving data and the label shall use plain text metadata which allows the data (or image in this case) to be used without additional processing. To ensure data in a TIFF/GeoTIFF file will be in a PDS4 acceptable form, these requirements must be followed:

- 1) Create a TIFF/GeoTIFF image that is compliant with version 6.0 of the TIFF standard.
- 2) No image compression is allowed.
- 3) Image data is a simple “raw” binary array (not tiled) compatible with Array_2D* or Array_3D*.
- 4) Multiple bands are allowed and the recommend layout should be band sequential (see Figure 1), although interleaved by line or pixel is allowed in PDS4 archives.
- 5) Colorized images can be supported using 3 bands (Array_3D*) and the typical Red, Green, and Blue channel order. Single band images with an inline color-palette should be avoided.
- 6) Within the PDS4 label pds:Header the “parsing_standard_id” should be listed as “TIFF 6.0” per the PDS4 schema.
- 7) The data should be formatted using the available bit types (see Table 1).

Table 1. TIFF/GeoTIFF allows the following data types:

TIFF Data Type (showing GDAL tag)	PDS4 Data Type
Byte	Unsigned Byte
UInt16	UnsignedLSB2 or UnsignedMSB2
Int16	SignedLSB2 or SignedMSB2
UInt32	UnsignedLSB4 or UnsignedMSB4
Int32	SignedLSB4 or SignedMSB4
Float32	IEEE754LSBSingle or IEEE754MSBSingle
Float64	IEEE754LSBDouble or IEEE754MSBDouble
CFloat32	ComplexLSB8 or ComplexMSB8
CFloat64	ComplexLSB16 or ComplexMSB16
Not supported	UnsignedLSB8 or UnsignedMSB8
<i>note: LSB or MSB is normally automatically selected depending on the host machine which created the TIFF image.</i>	

Convert to a PDS4-compliant TIFF/GeoTIFF using GDAL

The conversion from dozens of common image formats, including many planetary formats (e.g. ISIS3 cubes, VICAR, FITS, PDS3) to a PDS4-compliant format can be accomplished using the open source library Geospatial Data Abstraction Library (GDAL). The GDAL driver supports writing PDS4-compliant image data (e.g. Array_2D*, Array_3D*) as stored in a separate uncompressed raw file or an uncompressed GeoTIFF file (using a strip organization compatible with a raw file).

The GDAL routine, *gdal_translate*, works on all major platforms (Windows, Macintosh, and Linux). All image parameters like lines, samples, bands, bit type, machine type, and display direction will be automatically transferred to the created PDS4 exported image and PDS4 label. For PDS4 label entries that cannot be automated by the GDAL driver (e.g., mission name), a user-provided PDS4 XML template file with such metadata (and more) can be used during conversion. This allows the data provider to use existing PDS4 tools like the PDS Label Assistant for Interactive Design (PLAID) and On-Line Archiving Facility (OLAF) to create the initial PDS4 template. Also, user-defined variables within the provided template label

are supported and can be set during conversion (see example below). Lastly, the target body and radius, and when applicable, the map projection (e.g., Equirectangular), will be written to the Cartography section of the PDS4 label if supported in the source image.

An example PDS 4 conversion to the default “raw-only” output file would look like:

```
$ gdal_translate -of PDS4 -co TEMPLATE=in_pds4_template.xml in_image.cub
out_image_PDS4.xml
```

An example PDS 4 conversion to a “raw” GeoTIFF would similarly look like:

```
$ gdal_translate -of PDS4 -co IMAGE_FORMAT=GEOTIFF -co
TEMPLATE=in_pds4_template.xml in_image.fits out_PDS4.xml
```

An example PDS 4 label-only creation pointing into an existing PDS4-compliant TIFF/GeoTIFF image:

```
$ gdal_translate -of PDS4 -co IMAGE_FORMAT=EXISTING -co
TEMPLATE=in_pds4_template.xml pds4_image.tif out_PDS4.xml
```

A more complicated example using supported variables as defined in the input PDS4 template:

```
$ gdal_translate -of PDS4 input.tif output_PDS4.xml \
  -co VAR_TARGET_TYPE=Satellite \
  -co VAR_Target=Moon \
  -co VAR_OBSERVING_SYSTEM_NAME=LOLA \
  -co VAR_LOGICAL_IDENTIFIER=Lunar_LRO_LOLA_DEM_Global_64ppd.tif \
  -co VAR_TITLE="LRO LOLA Digital Elevation Model (DEM) 64ppd" \
  -co VAR_INVESTIGATION_AREA_NAME="Lunar Reconnaissance Orbiter" \
  -co VAR_INVESTIGATION_AREA_LID_REFERENCE= \
  "urn:nasa:pds:context:instrument_host:spacecraft.lro"
```

Convert to a PDS4-compliant TIFF using LibTIFF tools

Using the routines made available with the open source libtiff package, the conversion from a simple raw format to TIFF can be accomplished using *raw2tiff*. As a simple example, for an 8-bit raw image with 1440 samples and 720 lines this command can be used. Note “-H #” can be used to define how many bytes to skip for files with an inline header (e.g. PDS3, VICAR, FITS).

```
$ raw2tiff -w 1440 -l 720 -b 1 -H 0 -d byte in.raw output_PDS4.tif

Selected options:
-L          input data has LSB2MSB bit order (default)
-M          input data has MSB2LSB bit order
-H #       size of input image file header in bytes (0 by default)
-w #       width of input image in pixels
-l #       length of input image in lines
-b #       number of bands in input image (1 by default)

-d data_type  type of samples in input image
where data_type may be:
  byte       8-bit unsigned integer (default)
  short      16-bit unsigned integer
  long       32-bit unsigned integer
  ...
```

To use the output_PDS4.tif within a PDS4 archive you will still need to access the offset bytes for the inline TIFF header. As noted above, *gdal_translate* will write this value for you, but to find this value for an existing TIFF, the libtiff routine *tiffdump* can be used. For example:

```
$ tiffdump output_PDS4.tif
output_PDS4.tif:
Magic: 0x4949 <little-endian> Version: 0x2a <ClassicTIFF>
Directory 0: offset 8 (0x8) next 0 (0)
ImageWidth (256) SHORT (3) 1<1440>
ImageLength (257) SHORT (3) 1<720>
BitsPerSample (258) SHORT (3) 1<8>
Compression (259) SHORT (3) 1<1>
Photometric (262) SHORT (3) 1<1>
StripOffsets (273) LONG (4) 144<1252 8452 15652 22852 30052 37252 44452
51652 58852 66052 73252 80452 87652 94852 102052 109252 116452 123652 130852
138052 145252 152452 159652 166852 ...>
SamplesPerPixel (277) SHORT (3) 1<1>
...
```

The first value 1252 in the reported line “StripOffsets (273)” can be used within the PDS4 “offset” attribute as shown below.

```
...
  <Array_2D_Image>
    <local_identifier>image</local_identifier>
    <offset unit="byte">1252</offset>
    <axes>2</axes>
    <axis_index_order>Last Index Fastest</axis_index_order>
    <Element_Array>
      <data_type>UnsignedByte</data_type>
    </Element_Array>
...
```

Methods to Test for TIFF Compliance for a PDS4 Archive

There are several methods to test an existing TIFF/GeoTIFF for PDS4 compliance. One easy method is to simply run *gdal_translate* to create a PDS4 label using the mode `CREATE_LABEL_ONLY=YES` mode. If the file is compressed or tiled it will report that the TIFF is not PDS4 compliant. For example:

```
$ gdal_translate -of PDS4 -co ARRAY_TYPE=Array_2D_Image -co
CREATE_LABEL_ONLY=YES in_compressed.tif out_PDS4.lbl
Input file size is 1474593, 737297
ERROR 1: Source dataset is not compatible of a raw binary format
```

Using the GDAL utility *gdalinfo*, you can manually inspect a TIFF/GeoTIFF label for compliance. In the example below, both a compression is listed and a block size is defined denoting the image is tiled. Both of these format settings are not allowed within a PDS4 archive.

```
$ gdalinfo .\out.tif
Driver: GTiff/GeoTIFF
Size is 69154, 69154
...
Image Structure Metadata:
  COMPRESSION=LZW
  INTERLEAVE=BAND
...
Band 1 Block=256x256 Type=Byte, ColorInterp=Gray
```

A third method to check for compliance is to use the libtiff routine *tiffinfo*. Note that there might be several warnings reported for unknown TIFF tags which are often reported for GeoTIFFs. This is not a problem for a PDS4 archive. Below you will notice the reported Compression Scheme is “None” and there is no reported Tile Width or Tile Length, thus this image is valid for a PDS4 archive.

```
$ tiffinfo out.tif
...
TIFF Directory at offset 0x8 (8)
  Image Width: 1440 Image Length: 720
  Bits/Sample: 8
  Sample Format: unsigned integer
  Compression Scheme: None
  Photometric Interpretation: min-is-black
  Samples/Pixel: 1
  Rows/Strip: 5
  Planar Configuration: separate image planes
...
```

References:

- [1] The PDS4 Data Provider’s Handbook, Guide to Archiving Planetary Data Using the PDS4 Standard, 2020, Version 1.14.0 URL: <https://pds.nasa.gov/datastandards/documents/dph/>.
- [2] Hare, T.M., Rossi, A.P., Frigeri, A., Marmo, C., Interoperability in planetary research for geospatial data analysis. Planet. Space Sci. (2017). doi:10.1016/j.pss.2017.04.004. URL: <https://doi.org/10.1016/j.pss.2017.04.004>.
- [3] GeoTIFF, Revision 1.0, 2011, Sustainability of Digital Formats: Planning for Library of Congress Collections, URL: <https://www.loc.gov/preservation/digital/formats/fdd/fdd000279.shtml>.
- [4] OGC GeoTIFF Standard v1.1, 2019, Open Geospatial Consortium (OGC), URL: <http://docs.openeospatial.org/is/19-008r4/19-008r4.html>.
- [5] Electronic still-picture imaging - Part 2: TIFF/EP image data format, 2001, International Organization for Standardization (ISO), ISO:12234-2:2001, URL: <https://www.iso.org/obp/ui/#iso:std:iso:12234:-2:ed-1:v1:en>.
- [6] Graphic technology - Prepress digital data exchange - Tag image file format for image technology (TIFF/IT), 2004, International Organization for Standardization (ISO), ISO: ISO 12639:2004, URL: <https://www.iso.org/obp/ui/#iso:std:iso:12639:ed-2:v1:en>.
- [7] LibTIFF - TIFF Library and Utilities, 2003, version 3.6.1, URL: <http://www.libtiff.org/>.