NASA Planetary Data System (PDS) and GeoTIFF as a PDS4 Observational File

Prepared for the Planetary Data System by Trent Hare, thare@usgs.gov, October 2019

Introduction to GeoTIFF for PDS4

Here we introduce a proposal for the PDS to support the GeoTIFF format as an accepted “operational” image format (see p. 12 of the Data Providers Handbook). This builds on the precedent set by PDS4-compliant formats such as FITS and CDF. The GeoTIFF format was originally created in the late 1990’s 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.

As almost all mapping and image processing packages support GeoTIFF, it has emerged as a standard image file format worldwide (link). 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 (link). Also, in September 2019, GeoTIFF version 1.1 was adopted as an Open Geospatial Consortium (OGC) Standard (link). 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, ISO 12639). The GeoTIFF format is fully compliant with TIFF 6.0, so software incapable of reading and interpreting the specialized geospatial metadata will still be able to 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.

Technical Overview of 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. TIFF is a flexible file format for handling images with an associated descriptive header within a single file. While TIFF supports several compression methods and image tiling, it also supports the use of simple raw binary streams for image data, allowing TIFF to be **fully compliant with PDS4 requirements**. And although Adobe owns the copyright, there is no need for a license to implement 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, but any reader that supports ingestion of a raw binary stream can also read such a PDS4-compliant GeoTIFF format.

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. 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 used 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.

Discussion

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. Thus, in addition to being highly usable products, allowing for future archives to accept a PDS4-compliant TIFF or GeoTIFF file formats could cut archive data volumes in half, such that they do not need to support a duplicate copy within extras. Lastly, TIFF is already used for the Arizona State University’s Apollo Image Archive which scanned Apollo Metric and Panoramic Frame hard-copy film as raw TIFF 16-bit scans with PDS3 labels.

Summary

In summary, we strongly recommend that the TIFF/GeoTIFF format be allowed as a PDS4 “operational” approved data format. To support PDS4 compliance, we recommend that only the simplest TIFF format, essentially a raw binary stream for the image data, be allowed. The image data as stored in such a TIFF/GeoTIFF can easily be described in the PDS4 label (e.g., lines, samples, bands, skip bytes, bit type, display direction. etc.).

- GeoTIFF helps the PDS realize both a rigorously archival and an immediately usable format for the planetary science community (as PDS does with FITS for the astronomy community or CDF for the atmospheres community).
- While TIFF supports compression and tiling, the only form that would be allowed in a PDS4 archive would be a simple raw binary stream (easily accessible via the defined skip bytes).
- GeoTIFF is recognized as an archival format by the Library of Congress and supported by the Open Geospatial Consortium (OGC), an internationally supported standards group. Within the OGC supported 1.1 specification (link), there is an included lunar example, demonstrating its flexibility.
- Since 2017, the open source GDAL library (link) has included a robust implementation allowing for the creation of a PDS4-compliant TIFF/GeoTIFF, including checks to validate existing files.
- To realize this in PDS, we would need to add two attributes to the PDS4 schema “parsing_standard_id” (link): “TIFF/GeoTIFF", "TIFF/BigTIFF”, stating “The Header is governed by the Tagged Image File Format (TIFF) version 6.0 standard.”

Methods to Convert to a PDS4-compliant TIFF/GeoTIFF

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 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 additional this additional metadata, 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 last 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 out_PDS4.xml -co TEMPLATE=in_pds4_template.xml
in_image.cub
```

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

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

An example PDS 4 label creation pointing into an existing PDS4-compliant raw image:

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

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

```
$gdal_translate input.tif output.xml -of PDS4 \
   -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"
```