


OGC GeoTIFF Standard

Table of Contents

1. Scope	6
2. Conformance	7
3. References	8
4. Terms and Definitions	9
4.1. band	9
4.2. cell	9
4.3. code	9
4.4. coordinate	9
4.5. coordinate reference system	9
4.6. coordinate system	10
4.7. correspondence model	10
4.8. coverage	10
4.9. datum	10
4.10. device space	11
4.11. direct position	11
4.12. domain	11
4.13. double	11
4.14. ellipsoid	11
4.15. flattening	11
4.16. geocoding	12
4.17. geographic coordinate system	12
4.18. geokey	12
4.19. georectified grid	12
4.20. georeferencing	12
4.21. GeoTIFF	12
4.22. grid	13
4.23. grid	13
4.24. imagery	13
4.25. meridian	13
4.26. metadata	13
4.27. model space	14
4.28. mosaic	14
4.29. orthorectified grid	14
4.30. parallel	14
4.31. pixel	14
4.32. prime meridian	14
4.33. projected coordinate system	15
4.34. projection	15

4.35. raster	16
4.36. rational <TIFF>	16
4.37. rectified grid	16
4.38. referenceable grid	16
4.39. relative accuracy / relative positional accuracy	16
4.40. short	16
4.41. tag <TIFF>	16
5. Conventions	17
5.1. Abbreviations	17
5.2. ASCII	17
5.3. EPSG	17
5.4. IEEE	17
5.5. IFD	17
5.6. POSC	17
5.7. TIFF	17
5.8. USGS	17
5.9. Identifiers	17
6. Clauses not Containing Normative Material	18
7. Requirements	19
7.1. Underlying TIFF Requirements	19
7.1.1. Requirements Class TIFF	19
7.1.2. Requirements Class GeoKeyDirectoryTag	20
7.1.3. Requirements Class GeoKeyCode	22
7.1.4. Requirements Class GeoShortParams	22
7.1.5. Requirements Class GeoDoubleParamsTag	22
7.1.6. Requirements Class GeoAsciiParamsTag	22
7.2. Coordinate Transformation Requirements	23
7.2.1. Requirements Class ModelTiepointTag	23
7.2.2. Requirements Class ModelPixelScaleTag	24
7.2.3. Requirements Class ModelTransformationTag	24
7.2.4. Requirements Class IntergraphMatrixTag	24
7.3. GeoTIFF Configuration GeoKeys	24
7.3.1. Requirements Class GTModelTypeGeoKey	24
7.3.2. Requirements Class GTRasterTypeGeoKey	25
7.3.3. Requirements Class GTCitationGeoKey	26
7.4. Geographic CS Parameter GeoKeys	26
7.4.1. Requirements Class GeographicTypeGeoKey	26
7.4.2. Requirements Class GeogCitationGeoKey	27
7.4.3. Requirements Class GeogGeodeticDatumGeoKey	27
7.4.4. Requirements Class GeogPrimeMeridianGeoKey	28
7.4.5. Requirements Class GeogPrimeMeridianLongGeoKey	29

7.4.6. Requirements Class GeogLinearUnitsGeoKey	29
7.4.7. Requirements Class GeogLinearUnitSizeGeoKey	30
7.4.8. Requirements Class GeogAngularUnitsGeoKey	30
7.4.9. Requirements Class GeogAngularUnitSizeGeoKey	30
7.4.10. Requirements Class GeogEllipsoidGeoKey	31
7.4.11. Requirements Class GeogSemiMajorAxisGeoKey	31
7.4.12. Requirements Class GeogSemiMinorAxisGeoKey	32
7.4.13. Requirements Class GeogInvFlatteningGeoKey	32
7.4.14. Requirements Class GeogAzimuthUnitsGeoKey	33
7.5. Projected CS Parameter GeoKeys	33
7.5.1. Requirements Class ProjectedCSTypeGeoKey	33
7.5.2. Requirements Class PCSCitationGeoKey	34
7.6. Projection Definition GeoKeys	34
7.6.1. Requirements Class ProjectionGeoKey	34
7.6.2. Requirements Class ProjCoordTransGeoKey	35
7.6.3. Requirements Class ProjLinearUnitsGeoKey	35
7.6.4. Requirements Class ProjLinearUnitSizeGeoKey	36
7.6.5. Requirements Class ProjStdParallel1GeoKey	36
7.6.6. Requirements Class ProjStdParallel2GeoKey	37
7.6.7. Requirements Class ProjNatOriginLongGeoKey	37
7.6.8. Requirements Class ProjNatOriginLatGeoKey	37
7.6.9. Requirements Class ProjFalseEastingGeoKey	38
7.6.10. Requirements Class ProjFalseNorthingGeoKey	38
7.6.11. Requirements Class ProjFalseOriginLongGeoKey	38
7.6.12. Requirements Class ProjFalseOriginLatGeoKey	39
7.6.13. Requirements Class ProjFalseOriginEastingGeoKey	39
7.6.14. Requirements Class ProjFalseOriginNorthingGeoKey	40
7.6.15. Requirements Class ProjCenterLongGeoKey	40
7.6.16. Requirements Class ProjCenterLatGeoKey	40
7.6.17. Requirements Class ProjCenterEastingGeoKey	41
7.6.18. Requirements Class ProjCenterNorthingGeoKey	41
7.6.19. Requirements Class ProjScaleAtNatOriginGeoKey	41
7.6.20. Requirements Class ProjScaleAtCenterGeoKey	42
7.6.21. Requirements Class ProjAzimuthAngleGeoKey	42
7.6.22. Requirements Class ProjStraightVertPoleLongGeoKey	42
7.7. Vertical CS Parameter Keys	43
7.7.1. Requirements Class VerticalCSTypeGeoKey	43
7.7.2. Requirements Class VerticalCitationGeoKey	43
7.7.3. Requirements Class VerticalDatumGeoKey	43
7.7.4. Requirements Class VerticalUnitsGeoKey	44
8. Media Types for any data encoding(s)	45

Annex A: Conformance Class Abstract Test Suite (Normative)	46
A.1. Conformance Class A	46
A.1.1. Requirement 1	46
A.1.2. Requirement 2	46
Annex B: The GeoTIFF File Structure (Informative)	47
B.1. Introduction	47
B.2. GeoTIFF Design Considerations	47
B.3. GeoTIFF Software Requirements	48
B.4. GeoTIFF File and "Key" Structure	48
B.5. Coordinate Reference Systems in GeoTIFF	52
B.5.1. Device Space and GeoTIFF	52
B.5.2. Raster Coordinate Systems 	53
B.5.3. Model Coordinate Reference Systems	54
B.5.4. Reference Parameters	59
B.6. Coordinate Transformations	60
B.6.1. GeoTIFF Tags for Coordinate Transformations	60
B.6.2. Coordinate Transformation Data Flow	64
B.6.3. Cookbook for Defining Transformations	64
B.7. Geocoding Raster Data	65
B.7.1. General Approach	65
B.8. Examples	65
B.8.1. Common Examples	66
B.8.2. Less Common Examples	68
Annex C: Revision History	70
Annex D: Bibliography	71

Open Geospatial Consortium

Submission Date: <yyyy-mm-dd>

Approval Date: <yyyy-mm-dd>

Publication Date: <2018-12-16>

External identifier of this OGC® document: <http://www.opengis.net/doc/{doc-type}/{standard}/{m.n}>

Internal reference number of this OGC® document: YY-nnnrx

Version: 0.3

Category: OGC® <Implementation>

Editor: Ted Habermann

OGC GeoTIFF Standard

Copyright notice

Copyright © 2018 Open Geospatial Consortium

To obtain additional rights of use, visit <http://www.opengeospatial.org/legal/>

Warning

This document is not an OGC Standard. This document is distributed for review and comment. This document is subject to change without notice and may not be referred to as an OGC Standard.

Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type: OGC® Standard

Document subtype: Not Applicable

Document stage: Draft

Document language: English

License Agreement

Permission is hereby granted by the Open Geospatial Consortium, ("Licensor"), free of charge and subject to the terms set forth below, to any person obtaining a copy of this Intellectual Property and any associated documentation, to deal in the Intellectual Property without restriction (except as set forth below), including without limitation the rights to implement, use, copy, modify, merge, publish, distribute, and/or sublicense copies of the Intellectual Property, and to permit persons to whom the Intellectual Property is furnished to do so, provided that all copyright notices on the intellectual property are retained intact and that each person to whom the Intellectual Property is furnished agrees to the terms of this Agreement.

If you modify the Intellectual Property, all copies of the modified Intellectual Property must include, in addition to the above copyright notice, a notice that the Intellectual Property includes modifications that have not been approved or adopted by LICENSOR.

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD.

THE INTELLECTUAL PROPERTY IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE INTELLECTUAL PROPERTY WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE INTELLECTUAL PROPERTY WILL BE UNINTERRUPTED OR ERROR FREE. ANY USE OF THE INTELLECTUAL PROPERTY SHALL BE MADE ENTIRELY AT THE USER'S OWN RISK. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR ANY CONTRIBUTOR OF INTELLECTUAL PROPERTY RIGHTS TO THE INTELLECTUAL PROPERTY BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM ANY ALLEGED INFRINGEMENT OR ANY LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR UNDER ANY OTHER LEGAL THEORY, ARISING OUT OF OR IN CONNECTION WITH THE IMPLEMENTATION, USE, COMMERCIALIZATION OR PERFORMANCE OF THIS INTELLECTUAL PROPERTY.

This license is effective until terminated. You may terminate it at any time by destroying the Intellectual Property together with all copies in any form. The license will also terminate if you fail to comply with any term or condition of this Agreement. Except as provided in the following sentence, no such termination of this license shall require the termination of any third party end-user sublicense to the Intellectual Property which is in force as of the date of notice of such termination. In addition, should the Intellectual Property, or the operation of the Intellectual Property, infringe, or in LICENSOR's sole opinion be likely to infringe, any patent, copyright, trademark or other right of a third party, you agree that LICENSOR, in its sole discretion, may terminate this license without any compensation or liability to you, your licensees or any other party. You agree upon termination of any kind to destroy or cause to be destroyed the Intellectual Property together with all copies in any form, whether held by you or by any third party.

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize

you or any third party to use certification marks, trademarks or other special designations to indicate compliance with any LICENSOR standards or specifications. This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it.

i. Abstract

This profile specifies the requirements and encoding rules for using the Tagged Image File Format (TIFF) for the exchange of georeferenced imagery. It formalizes the existing community standard for the Geographic Tagged Image File Format (GeoTIFF) file format.

ii. Keywords

The following are keywords to be used by search engines and document catalogues.

ogcdoc, OGC document, geotiff, tiff

iii. Preface

The GeoTIFF format was initially developed during the early 1990's (N. Ritter & Ruth, 1997) in order to leverage a mature platform independent file format (TIFF) by adding metadata required for describing and using geographic image data. TIFF met the requirements for an underlying format, as it was lossless and extensible. In September 1994, SPOT Image Corp proposed a GeoTIFF structure that was limited to Universal Transverse Mercator (N. Ritter & Ruth, 1997). The proposed GeoTIFF specification has augmented and formalized by Niles and Ruth as Revision 1.0, specification version 1.8.2 in November 1995 (N. Ritter & Ruth, 1995). This specification is currently the official GeoTIFF specification (GeoTIFF, n.d).

The GeoTIFF format is used throughout the geospatial and earth science communities to share geographic image data. That usage inevitably leads to identification of new requirements and needs for profiles, extensions, and improvements to the original GeoTIFF Specification. The OGC is well established as a forum for standardization in the GeoTIFF producer and user communities and, as such, it provides an inclusive standardization process for those communities. This document is the first step in the process of integration of the GeoTIFF into that standardization process. Once GeoTIFF is ensconced in the OGC, the standard can be evolved using a formal process.

Suggested additions, changes, and comments on this standard are welcome and encouraged. Such suggestions may be submitted by email message or by submitting an official OGC Change Request using the online CR application:

https://portal.opengeospatial.org/public_ogc/change_request.php

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

iv. Submitting organizations

The following organizations submitted this Document to the Open Geospatial Consortium (OGC):

The HDF Group

v. Submitters

All questions regarding this submission should be directed to the editor or the submitters:

Name	Affiliation
Ted Habermann	The HDF Group
Emmanuel Devys	IGN
Charles Heazel	Heazeltech LLC

Chapter 1. Scope

This OGC® Standard defines a set of TIFF tags provided to describe all "Cartographic" information associated with TIFF imagery that originates from satellite imaging systems, scanned aerial photography, scanned maps, digital elevation models, or as a result of geographic analyses. Its aim is to allow means for tying a raster image to a known model space or map projection and for describing those projections. This OGC® Standard defines the Geographic Tagged Image File Format (GeoTIFF) file format and the requirements to which every GeoTIFF file must adhere. It focuses on updating the current GeoTIFF community specification and aligning it with current OGC standardization practice.

The tags documented in this spec are to be considered completely orthogonal to the raster-data descriptions of the TIFF spec, and impose no restrictions on how the standard TIFF tags are to be interpreted, which color spaces or compression types are to be used, etc.

Chapter 2. Conformance

This standard defines 120 Requirements grouped into 29 Requirements Classes.

These Requirements address five (5) standardization target types:

- Underlying TIFF Requirements
- GeoTIFF Configuration GeoKeys
- Geographic CS Parameter GeoKeys
- Projected CS Parameter GeoKeys
- Vertical CS Parameter GeoKeys

Conformance with this standard shall be checked using all the relevant tests specified in Annex A (normative) of this document. The framework, concepts, and methodology for testing, and the criteria to be achieved to claim conformance are specified in the OGC Compliance Testing Policies and Procedures and the OGC Compliance Testing web site.

In order to conform to this OGC® encoding standard, a software implementation shall comply with all of the conformance classes specified in Annex A (normative).

This version 1.0 of the OGC® GeoTIFF standard is a formalized version of the GeoTIFF specification currently in common use. Any discrepancies between this specification and the community GeoTIFF standard identified in Chapter 3 should be brought to the attention of the OGC® for resolution.

Chapter 3. References

The following normative documents contain provisions that, through reference in this text, constitute provisions of this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the normative document referred to applies.

- Association Adobe Developers, *TIFF Specification Revision 6.0*, June 3, 1992
- Ritter, Niles & Ruth, Mike, *GeoTIFF Format Specification*, October 31, 1995

Chapter 4. Terms and Definitions

This document uses the terms defined in Sub-clause 5.3 of [OGC 06-121r8], which is based on the ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards. In particular, the word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this standard.

For the purposes of this document, the following additional terms and definitions apply.

4.1. band

well-defined range of wavelengths, frequencies or energies of optical, electric or acoustic radiation

Note 1 to entry: At the pixel level, a band is represented as one of the vector values of the pixel. At image level, band i of an image is the rectangular array of i^{th} sample values from the pixel vectors.

[Source: 19101-2:2008, 4.1]

4.2. cell

rectangular area in raster space, in which a single pixel value is filled

4.3. code


representation of a label according to a specified scheme

[Source: ISO 19118:2011, 4.3]

4.4. coordinate

one of a sequence of numbers designating the position of a point


Note 1 to entry: In a spatial coordinate reference system, the coordinate numbers are qualified by units.

[Source: ISO 19111 Rev 2017-12-17 in progress 


4.5. coordinate reference system

coordinate system that is related to an object (of the real world)  a datum

Note 1 to entry: Geodetic and vertical datums are referred to as reference frames.


Note 2 to entry: For geodetic and vertical reference frames, the object will be the Earth 

Note 3 to entry: In planetary applications geodetic reference frames may be applied to other celestial bodies.

[Source: ISO 19111 Rev 2017-12-17 in progress 

4.6. coordinate system

set of mathematical rules for specifying how coordinates are to be assigned to points

[Source: ISO 19111 Rev 2017-12-17 in progress 

4.7. correspondence model

functional relationship between ground and image coordinates based on the correlation between a set of ground control points and their corresponding image coordinates

[Source: ISO/TS 19130:2010, 4.3]

4.8. coverage

feature that acts as a function to return values from its range for any direct position within its spatial, temporal, or spatiotemporal domain


Example: Examples include a digital image, raster map, and digital elevation matrix.


Note 1 to entry: In other words, a coverage is a feature that has multiple values for each attribute type, where each direct position within the geometric representation of the feature has a single value for each attribute type.

[Source: ISO 19123:2005, 4.1.7]

4.9. datum

reference frame

parameter or set of parameters that define the position of the origin, the scale, and the orientation of a coordinate system 

[Source: ISO 19111 Rev 2017-12-17 in progress 

4.10. device space

coordinate space referencing scanner, printers and display devices

4.11. direct position

position described by a single set of coordinates within a coordinate reference system

4.12. domain

well-defined set.

Note 1 to entry: Domains are used to define the domain set and range set of operators and functions.

Note 2 to entry: Well-defined means that the definition is both necessary and sufficient, as everything that satisfies the definition is in the set and everything that does not satisfy the definition is necessarily outside the set.


[Source: 19109:2015, 4.8]


4.13. double

8-byte IEEE double precision floating point


4.14. ellipsoid

reference ellipsoid

surface formed by the rotation of an ellipse about a main axis 

<geodesy> geometric reference surface embedded in 3D Euclidean space represented by an ellipsoid 

Note 1 to entry: For the Earth the ellipsoid is bi-axial with rotation about the polar axis. This results in an oblate ellipsoid with the midpoint of the foci located at the nominal center  the Earth.


[Source: ISO 19111 Rev 2017-12-17, in progress] 

4.15. flattening

ratio of the difference between the semi-major (a) and semi-minor axis (b) of an ellipsoid to the semi-major axis; $f = (a - b)/a$

Note 1 to entry: Sometimes inverse flattening $1/f = a/(a - b)$ is given instead; $1/f$ is also known as


reciprocal flattening.

[Source: ISO 19111 Rev 2017-12-17 in progress] 

4.16. geocoding

translation of one form of location into another

4.17. geographic coordinate system

coordinate system consisting of a well-defined ellipsoidal datum, a Prime Meridian, and an angular unit, allowing the assignment of a Latitude-Longitude (and optionally, geodetic height) vector to a location on earth 

4.18. geokey

in GeoTIFF, a GeoKey is equivalent in function to a TIFF tag, but uses a different storage mechanism

4.19. georectified grid

rectified grid

grid for which there is an affine transformation between the grid coordinates and the coordinates of an external coordinate reference system

Note 1 to entry: If the coordinate reference system is related to the earth by a datum, the grid is a georectified grid.

[Source: ISO 19123:2005, 4.1.32]

4.20. georeferencing

geopositioning an object using a Correspondence Model derived from a set of points for which both ground and image coordinates are known

[Source: ISO 19130:2010, 4.37]


4.21. GeoTIFF

standard for storing georeference and geocoding information in a TIFF 6.0 compliant raster file

4.22. grid

network composed of two or more sets of curves in which the members of each set intersect the members of the other sets in an algorithmic way

4.23. grid

gridded data Network composed of two or more sets of curves in which the members of each set intersect the members of the other sets in a algorithmic way. 

Note 1 to entry: The curves partition a space into grid cells.

[Source: ISO 19123:2005, 4.1.23]

4.24. imagery

representation of phenomena as images produced electronically and/or optical techniques

Note 1 to entry: In this document, it is assumed that the phenomena have been sensed or detected by one or more devices such as radar, cameras, photometers, and infra-red and multispectral scanners


[Source: 19101-2:2008, 4.14]

4.25. meridian

An intersection of an ellipsoid by a plane containing the shortest axis of the ellipsoid

 An intersection of an ellipsoid by a plane containing the shortest axis of the ellipsoid

Note 1 to entry: This term is generally used to describe the pole-to-pole arc rather than the complete closed figure.


[Source: 19111 Rev. 2017-12-17 in progress] 

4.26. metadata

information about a resource

[Source: ISO 19115-1:2014, 4.10]

4.27. model space

flat geometrical space used to model a portion of the earth 

4.28. mosaic

an image composed of two or more separately collected (sensed) images

Note 1 to entry: Additional XML metadata may be used to identify the cut-lines (boundaries and parameters for the images used to compose the mosaic.

4.29. orthorectified grid

orthoimage

georectified grid created using ground control points and elevation data where constant scale is maintained throughout the grid

image in which by orthogonal projection to a reference surface, displacement of image points due to sensor orientation and terrain relief has been removed

Note 1 to entry: The amount of displacement depends on the resolution and the level of detail of the elevation information and on the software implementation.

4.30. parallel

line  of constant latitude, parallel to the equator

4.31. pixel

smallest element of a digital image to which attributes are assigned.


Note 1 This term originated as a contraction of “picture element”.

Note 2 Related to the concept of a grid cell.

Note 3 to entry: The intensity of each pixel is variable; in color systems, each pixel has typically three or four dimensions of variability such as red, green and blue, or cyan, magenta, yellow and black.


4.32. prime meridian


meridian from which the longitudes of other meridians are quantified

[Source: ISO 19111 Rev 2017-12-17 in progress] 

4.33. projected coordinate system


projected coordinate reference system 

coordinate reference system derived from a two-dimensional geodetic coordinate reference system by applying a map projection 

projected coordinate reference system coordinate reference system derived from a geographic coordinate reference system by applying a map projection 

Note 1 to entry: May be two- or three-dimensional, the dimension being equal to that of the geographic coordinate reference system from which it is derived.

Note 2 to entry: In the three-dimensional case the horizontal coordinates (geodetic latitude and geodetic longitude coordinates) are projected to northing and easting and the ellipsoidal height is unchanged.

[Source ISO 19111 Rev 2017-12-17, in progress] 


4.34. projection


projected coordinate reference system

coordinate reference system derived from a two-dimensional geodetic coordinate reference system by applying a map projection projected coordinate reference system

coordinate reference system derived from a geographic coordinate reference system by applying a map projection

Note 1 to entry: May be two- or three-dimensional, the dimension being equal to that of the geographic coordinate reference system from which it is derived.

Note 2 to entry: In the three-dimensional case the horizontal coordinates (geodetic latitude and geodetic longitude coordinates) are projected to northing and easting and the ellipsoidal height is unchanged. 

[Source: ISO 19111 Rev 2017-12-17, in progress] 

range <coverage> 

set of feature attribute values associated by a function with the elements of the domain of a coverage

[Source: ISO 19123:2005, 4.1.29]

4.35. raster

raster space

usually rectangular pattern of parallel scanning lines forming or corresponding to the display on a cathode ray tube

Note 1 to entry: A raster is a type of grid.

[Source: ISO 19123:2005, 4.1.30]

4.36. rational <TIFF>

a RATIONAL value is a fractional value represented by the ratio of two unsigned 4-byte integers

4.37. rectified grid

grid for which there is an affine transformation between the grid coordinates and the coordinates of an external coordinate reference system

4.38. referenceable grid

grid associated with a transformation that can be used to convert grid coordinate values to values of coordinates referenced to an external coordinate reference system

4.39. relative accuracy / relative positional accuracy

evaluation of the random errors in determining the position of one point or feature with respect to another / closeness of coordinate difference value to the true or accepted value in a specified reference system

4.40. short

2-byte IEEE signed integer

4.41. tag <TIFF>

a tag is packet of numerical or ASCII values, which have a numerical "Tag" ID indicating their information content




Chapter 5. Conventions

5.1. Abbreviations

5.2. ASCII

[American Standard Code for Information Interchange] The predominant character set encoding of present-day computers.

5.3. EPSG

European Petroleum Survey Group 

5.4. IEEE

Institute of Electrical and Electronics Engineers, Inc.

5.5. IFD

In TIFF format, an Image File Directory, containing all the TIFF tags for one image in the file (there may be more than one).

5.6. POSC

Petrotechnical Open Software Corporation.

5.7. TIFF

Acronym for Tagged Image File Format; a platform-independent, extensive specification for storing raster data and ancillary information in a single file.

5.8. USGS

United States Geological Survey

5.9. Identifiers

The normative provisions in this specification are denoted by the URI

<http://www.opengis.net/spec/{standard}/{m.n}>

All requirements and conformance tests that appear in this document are denoted by partial URIs which are relative to this base.

Chapter 6. Clauses not Containing Normative Material

The following section of this Standard is included for informative purposes only. It serves to assist in correct implementation of this standard but do not impose any additional requirements.

- **Annex B: The GeoTIFF File Structure:** This Annex provides a **discription** of the community GeoTIFF specification and supporting information on how various elements of that specification should be used.



Chapter 7. Requirements

This OGC Standard specifies the content and structure of a group of industry-standard tag sets for the management of **georeferenced** geocoded raster imagery using Aldus-Adobe's public domain Tagged-Image File Format (TIFF).

This standard defines a set of TIFF tags provided to describe all "Cartographic" information associated with TIFF imagery that originates from satellite imaging systems, scanned aerial photography, scanned maps, digital elevation models, or as a result of geographic analyses. Its aim is to allow means for tying a raster image to a known model space **or map projection**, and for describing **those projections**.

7.1. Underlying TIFF Requirements

This Standardization Target addresses the core Requirements Classes applicable to all GeoTIFF files.

A GeoTIFF file is a TIFF 6.0 file and inherits the file structure as described in the corresponding portion of the TIFF spec. In addition, all GeoTIFF specific information is encoded in several reserved TIFF tags. The following requirements formalize compliance with TIFF and the GeoTIFF reserved tag structure.

The tags documented in this standard are to be considered completely orthogonal to the raster-data descriptions of the TIFF spec, and impose no restrictions on how the standard TIFF tags are to be interpreted, which color spaces or compression types are to be used, etc.

7.1.1. Requirements Class TIFF

A GeoTIFF file is a valid TIFF 6.0 file.

Requirements Class: TIFF	
	http://www.opengis.net/spec/GeoTIFF/0.0/Core
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/TIFF <i>A GeoTIFF file SHALL be compliant with the TIFF 6.0 specification</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/DataGeoTags <i>GeoTIFF files SHALL encode all GeoTIFF specific information using the specified reserved TIFF tags</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/DataTypes <i>GeoTIFF implementation software SHALL support all documented TIFF 6.0 tag data-types, and in particular the IEEE double-precision floating point "DOUBLE" type</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ByteOrder <i>GeoTIFF implementation software SHALL honor the 'byte-order' indicator by performing byte swaps as necessary to provide an accurate in-memory representation of the values encoded in the GeoTIFF file</i>

Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/TagSort <i>The TIFF tags in a GeoTIFF file SHALL be written out to the file with the tag-IDs sorted in ascending order</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeySort <i>The GeoKey entries in a GeoTIFF file SHALL be written out to the file with the key-IDs sorted in ascending order</i>

7.1.2. Requirements Class GeoKeyDirectoryTag

The GeoKeyDirectoryTag Requirements Class specifies the requirements for implementing the reserved GeoKeyDirectoryTag TIFF tag.

A GeoTIFF file stores projection parameters in a set of "Keys" which are virtually identical in function to a TIFF tag, but have one more level of abstraction above TIFF. Like a tag, a Key has an ID number ranging from 0 to 65535, but unlike TIFF tags, all key ID's are available for use in GeoTIFF parameter definitions.

The Keys in GeoTIFF (also call "GeoKeys") are all referenced from the GeoKeyDirectoryTag tag. The first four keys form the GeoKey Directory Header. The keys which make up this header are; KeyDirectoryVersion, KeyRevision, MinorRevision, and NumberOfKeys.

The GeoKey Directory Header is followed by <NumberOfKeys> Key Entries. Each Key Entry consists of four values:

- "KeyID" gives the key-ID value of the Key (identical in function to TIFF tag ID, but completely independent of TIFF tag-space).
- "TIFFTagLocation" indicates which TIFF tag contains the value(s) of the Key: if TIFFTagLocation is 0, then the value is SHORT, and is contained in the "Value_Offset" entry. Otherwise, the type (format) of the value is implied by the TIFF-Type of the tag containing the value.
- "Count" indicates the number of values in this key.
- "Value_Offset" Value_Offset indicates the index- offset **into** the TagArray indicated by TIFFTagLocation, if it is nonzero. If TIFFTagLocation=0, then Value_Offset contains the actual (SHORT) value of the Key, and Count=1 is implied. Note that the offset is not a byte-offset, but rather an index based on the natural data type of the specified tag array.

Requirements Class: GeoKeyDirectoryTag	
http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.ID <i>The GeoKeyDirectoryTag SHALL have ID = 34735</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.type <i>The GeoKeyDirectoryTag SHALL have type = SHORT (2-byte unsigned integer)</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.count <i>The GeoKeyDirectoryTag SHALL include at least 4 keys (short integers) as header information</i>

Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.keyDirectoryVersion <i>The first unsigned short integer in the GeoKeyDirectoryTag SHALL hold the KeyDirectoryVersion.</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.keyRevision <i>The second unsigned short integer in the GeoKeyDirectoryTag SHALL hold the KeyRevision.</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.minorRevision <i>The third unsigned short integer in the GeoKeyDirectoryTag SHALL hold the MinorRevision.</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.minorRevisionValue <i>The MinorRevision for this standard SHALL be 1.0.</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.numberOfKeys <i>The fourth unsigned short integer in the GeoKeyDirectoryTag SHALL hold the NumberOfKeys defined in the rest of the GeoKeyDirectoryTag.</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.keyEntrySetCount <i>The GeoKeyDirectoryTag SHALL hold NumberOfKeys KeyEntry Sets in addition to the header information</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.KeyEntry <i>Each Key Entry in the Key Entry Set SHALL include 4 unsigned short integer values</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.keyEntryKeyID <i>The first unsigned short integer in the Key Entry SHALL hold the key identifier.</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.keyEntryTIFFTagLocation <i>The second unsigned short integer in the Key Entry SHALL hold the TIFF Tag Location. The value of this entry shall be a valid GeoTIFF tag identifier or a zero (0)</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.keyEntryKeyCount <i>The Third unsigned short integer in the Key Entry SHALL indicate the number of values associated with this key.</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyDirectoryTag.keyEntryValueOffset <i>The fourth unsigned short integer in the Key Entry SHALL hold either the key value (if TIFF Tag location = 0) or the index into the tag indicated by the TIFF Tag Location value.</i>

7.1.3. Requirements Class GeoKeyCode

For consistency, several key codes have the same meaning in all implemented GeoKeys

*Requirements Class: GeoKeyCode	
http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyCode	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyCode.undefined <i>GeoKeys with a value of zero SHALL indicate intentionally omitted parameters</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoKeyCode.userDefined <i>GeoKeys with a value of 32767 SHALL indicate user-defined parameters</i>

The "undefined" code means that this parameter is intentionally omitted.

In some cases, additional GeoKeys are required when the "User-Defined" is used. Those requirements are included within a Requirements Class where appropriate.

7.1.4. Requirements Class GeoShortParams

The following requirements govern the storage of parameter values when there are two or more values of type Short.

Requirements Class: GeoShortParams	
http://www.opengis.net/spec/GeoTIFF/0.0/GeoShortParams	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoShortParams.Criteria <i>In the case where a Parameter of type Short has more than one value, those values SHALL be stored in the GeoKeysDirectory</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoShortParams.Location <i>Parameter values stored in the GeoKeysDirectory SHALL appear after the last Key Entry</i>

7.1.5. Requirements Class GeoDoubleParamsTag

The following requirements govern the storage of parameter values when the values are of type Double.

Requirements Class: GeoDoubleParamsTag	
http://www.opengis.net/spec/GeoTIFF/0.0/GeoDoubleParamsTag	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoDoubleParamsTag.ID <i>The GeoDoubleParamsTag SHALL have ID = 34736</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoDoubleParamsTag.count <i>The GeoDoubleParamsTag MAY hold any number of key parameters with type = Double.</i>


7.1.6. Requirements Class GeoAsciiParamsTag

The following requirements govern the storage of parameter values when the values are of type

ASCII. All text values in a TIFF file must be null-terminated ASCII.

Requirements Class: GeoAsciiParamsTag	
http://www.opengis.net/spec/GeoTIFF/0.0/GeoAsciiParamsTag	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoAsciiParamsTag.ID <i>The GeoAsciiParamsTag SHALL have ID = 34737</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoAsciiParamsTag.count <i>The GeoAsciiParamsTag MAY hold any number of key parameters with type = ASCII. (May not be necessary, the same as keyentry.count)</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoAsciiParamsTag.NULLRead <i>Pipe characters (" ") in the GeoAsciiParamsTag SHALL be converted NULL characters before returning strings to the client</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoAsciiParamsTag.NULLWrite <i>NULL characters used to terminate strings in the GeoAsciiParamsTag SHALL be converted to a " " (pipe) prior to being written into the GeoAsciiParamsTag</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeoAsciiParamsTag.type <i>The GeoAsciiParamsTag SHALL have type = ASCII</i>

7.2. Coordinate Transformation Requirements

For most common applications, the transformation between raster  and model space may be defined with a set of raster-to-model tiepoints and scaling parameters. The ModelTiepointTag and ModelPixesScaleTag may be used for this purpose.

Alternatively, the ModelTransformationTag may be used to specify the transformation matrix between the raster space (and its dependent pixel-value space) and the (possibly 3D) model space.

The IntergraphMatrixTag has been deprecated and should not be used. It is included in this version of the GeoTIFF standard for completeness.

7.2.1. Requirements Class ModelTiepointTag

Requirements Class: ModelTiepointTag	
http://www.opengis.net/spec/GeoTIFF/0.0/ModelTiepointTag	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ModelTiepointTag.ID <i>The ModelTiepointTag SHALL have ID = 33922</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ModelTiepointTag.type <i>The ModelTiepointTag SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ModelTiepointTag.count <i>The ModelTiepointTag SHALL have 6 values for each of the K tiepoints</i>

7.2.2. Requirements Class ModelPixelScaleTag

Requirements Class: ModelPixelScaleTag	
http://www.opengis.net/spec/GeoTIFF/0.0/ModelPixelScaleTag	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ModelPixelScaleTag.ID <i>The ModelPixelScaleTag SHALL have ID = 33550</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ModelPixelScaleTag.type <i>The ModelPixelScaleTag SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ModelPixelScaleTag.count <i>The ModelPixelScaleTag SHALL have 3 values representing the scale factor in the X, Y, and Z directions</i>

7.2.3. Requirements Class ModelTransformationTag

Requirements Class: ModelTransformationTag	
http://www.opengis.net/spec/GeoTIFF/0.0/ModelTransformationTag	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ModelTransformationTag.ID <i>The ModelTransformationTag SHALL have ID = 34264</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ModelTransformationTag.type <i>The ModelTransformationTag SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ModelTransformationTag.count <i>The ModelTransformationTag SHALL have 16 values representing the terms of the 4 by 4 transformation matrix. The terms SHALL be in row-major order</i>

7.2.4. Requirements Class IntergraphMatrixTag

Unresolved directive in clause_7_requirements.adoc -
include::requirements/Model_Tags/requirements_class_ItergraphMatrixTag.adoc[]

7.3. GeoTIFF Configuration GeoKeys


This Standardization Target addresses encoding of Configuration values essential for interpreting the rest of the GeoKeys.




7.3.1. Requirements Class GTModelTypeGeoKey

This GeoKey defines the general type of model Coordinate system used, and to which the raster space will be transformed: unknown, Geocentric (rarely used), Geographic, Projected Coordinate System, or user-defined. If the coordinate system is a PCS, then only the PCS code need be specified. If the coordinate system does not fit into one of the standard registered PCS'S, but it uses one of the standard projections and datums, then its should be documented as a PCS model with "user-defined" type, requiring the specification of projection parameters, etc.

Requirements Class: GTModelTypeGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GTModelTypeGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTModelTypeGeoKey.ID <i>The GTModelTypeGeoKey SHALL have ID = 1024</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTModelTypeGeoKey.type <i>The GTModelTypeGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTModelTypeGeoKey.projected <i>A value of 1 for the GTModelTypeGeoKey SHALL indicate a projected coordinate system</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTModelTypeGeoKey.geographic <i>A value of 2 for the GTModelTypeGeoKey SHALL indicate a geographic latitude-longitude coordinate system</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTModelTypeGeoKey.geocentric <i>A value of 3 for the GTModelTypeGeoKey SHALL indicate a geocentric(X,Y,Z) coordinate system</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTModelTypeGeoKey.reserved <i>GTModelTypeGeoKey values in the range 4-32766 SHALL be reserved</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTModelTypeGeoKey.user-defined <i>If the GTModelTypeGeoKey value is 32767 (User-Defined) then the GTCitationGeoKey SHALL be populated</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTModelTypeGeoKey.private <i>GTModelTypeGeoKey values in the range 32768-65535 SHALL be private</i>

7.3.2. Requirements Class GTRasterTypeGeoKey

This establishes the Raster Space **coordinate system** d; there are currently only two, namely RasterPixelIsPoint and RasterPixelIsArea. No user-defined raster spaces are currently supported. For variance in imaging display parameters, such as pixel aspect-ratios, use the standard TIFF 6.0 device-space tags instead.

Requirements Class: GTRasterTypeGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GTRasterTypeGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTRasterTypeGeoKey.ID <i>The GTModelTypeGeoKey SHALL have ID = 1025</i> 
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTRasterTypeGeoKey.type <i>The GTModelTypeGeoKey SHALL have type = SHORT</i> 
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTRasterTypeGeoKey.reserved <i>GTRasterTypeGeoKey values in the range 1-32766 SHALL be reserved</i> 

Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTRasterTypeGeoKey.private <i>GTRasterTypeGeoKey values in the range 32768-65535 SHALL be private</i>
-------------	---

7.3.3. Requirements Class GTCitationGeoKey

As with all the "Citation" GeoKeys, this is provided to give an ASCII reference to published documentation on the overall configuration of this GeoTIFF file.

Requirements Class: GTCitationGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GTCitationGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTCitationGeoKey.ID <i>The GTCitationGeoKey SHALL have ID = 1026</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GTCitationGeoKey.type <i>The GTCitationGeoKey SHALL have type = ASCII</i>

7.4. Geographic CS Parameter GeoKeys

In general, the geographic coordinate system used will be implied by the projected coordinate system code. If however, this is a user-defined PCS, or the ModelType was chosen to be Geographic, then the system must be explicitly defined here, using the Horizontal datum code.

7.4.1. Requirements Class GeographicTypeGeoKey

This key may be used to specify the code for the geographic coordinate system used to map lat-long to a specific ellipsoid over the earth.

Requirements Class: GeographicTypeGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeographicTypeGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeographicTypeGeoKey.ID <i>The GeographicTypeGeoKey SHALL have ID = 2048</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeographicTypeGeoKey.type <i>The GeographicTypeGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeographicTypeGeoKey.obsolete <i>GeographicTypeGeoKey values in the range 1-1000 SHALL be obsolete EPSG/POSC Geographic Codes</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeographicTypeGeoKey.EPSGGeographic <i>GeographicTypeGeoKey values in the range 1024-32766 SHALL be geographic 2D or 3D EPSG CRS codes</i> NOTE: Original ranges were 4000-4199 and 4200-4999

Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeographicTypeGeoKey.user-defined If the <i>GTGeographicTypeGeoKey</i> value is 32767 (User-Defined) then the <i>GeogCitationGeoKey</i> , <i>GeogGeodeticDatumGeoKey</i> , <i>GeogAngularUnitsGeoKey</i> , and <i>GeogPrimeMeridianGeoKey</i> SHALL be populated
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeographicTypeGeoKey.private <i>GeographicTypeGeoKey</i> values in the range 32768-65535 SHALL be private

7.4.2. Requirements Class *GeogCitationGeoKey*

General citation and reference for all Geographic CS parameters.

Requirements Class: <i>GeogCitationGeoKey</i>	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogCitationGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogCitationGeoKey.ID The <i>GeogCitationGeoKey</i> SHALL have ID = 2049
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogCitationGeoKey.type The <i>GeogCitationGeoKey</i> SHALL have type = ASCII

7.4.3. Requirements Class *GeogGeodeticDatumGeoKey*

This key may be used to specify the horizontal datum, defining the size, position and orientation of the reference ellipsoid used in user-defined geographic coordinate systems.

Requirements Class: <i>GeogGeodeticDatumGeoKey</i>	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogGeodeticDatumGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogGeodeticDatumGeoKey.ID The <i>GeogGeodeticDatumGeoKey</i> SHALL have ID = 2050
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogGeodeticDatumGeoKey.type The <i>GeogGeodeticDatumGeoKey</i> SHALL have type = SHORT
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogGeodeticDatumGeoKey.obsolete <i>GeogGeodeticDatumGeoKey</i> values in the range 1-1000 SHALL be obsolete EPSG/POSC Datum Codes
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogGeodeticDatumGeoKey.reserved <i>GeogGeodeticDatumGeoKey</i> values in the ranges 1001-5999 or 7000-32766 SHALL be reserved by GeoTIFF
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogGeodeticDatumGeoKey.EPSGEllipsoid <i>GeogGeodeticDatumGeoKey</i> values in the range 6000-6199 SHALL be EPSC datum codes based on ellipsoid only

Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogGeodeticDatumGeoKey.EPSGDatum <i>GeogGeodeticDatumGeoKey values in the range 6200-6999 SHALL be EPSC datum codes based on EPSG datum</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogGeodeticDatumGeoKey.WGSDatum <i>GeogGeodeticDatumGeoKey values in the range 6322-6327 SHALL be code for WGS datum</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogGeodeticDatumGeoKey.ArchaicDatum <i>GeogGeodeticDatumGeoKey values in the range 6900-6999 SHALL be codes for archaic datum</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogGeodeticDatumGeoKey.user-defined <i>If the GeogGeodeticDatumGeoKey value is 32767 (User-Defined) then the GeogCitationGeoKey, and GeogEllipsoidGeoKey SHALL be populated</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogGeodeticDatumGeoKey.private <i>GeogGeodeticDatumGeoKey values in the range 32768-65535 SHALL be private</i>

7.4.4. Requirements Class GeogPrimeMeridianGeoKey

Allows specification of the location of the Prime meridian for user-defined geographic coordinate systems. The default standard is Greenwich, England.

Requirements Class: GeogPrimeMeridianGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogPrimeMeridianGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogPrimeMeridianGeoKey.ID <i>The GeogPrimeMeridianGeoKey SHALL have ID = 2051</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogPrimeMeridianGeoKey.type <i>The GeogPrimeMeridianGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogPrimeMeridianGeoKey.obsolete <i>GeogPrimeMeridianGeoKey values in the range 1-100 SHALL be obsolete EPSG/POSC Datum Codes</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogPrimeMeridianGeoKey.reserved <i>GeogPrimeMeridianGeoKey values in the range 101-1023 SHALL be reserved</i>
NOTE: Original range was 101-7999 and 9000-32766	

Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogPrimeMeridianGeoKey.EPSGEllipsoid <i>GeogPrimeMeridianGeoKey values in the range 1024-32766 SHALL be EPSG Prime Meridian Codes</i> NOTE: Original range was 8000-8999
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogPrimeMeridianGeoKey.private <i>GeogPrimeMeridianGeoKey values in the range 32768-65535 SHALL be private</i>

7.4.5. Requirements Class GeogPrimeMeridianLongGeoKey

This key allows definition of user-defined Prime Meridians, the location of which is defined by its longitude relative to Greenwich.

Requirements Class: GeogPrimeMeridianLongGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogPrimeMeridianLongGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogPrimeMeridianLongGeoKey.ID <i>The GeogPrimeMeridianLongGeoKey SHALL have ID = 2061</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogPrimeMeridianLongGeoKey.type <i>The GeogPrimeMeridianLongGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogPrimeMeridianLongGeoKey.units <i>The GeogPrimeMeridianLongGeoKey SHALL have units = GeogAngularUnits</i>

7.4.6. Requirements Class GeogLinearUnitsGeoKey

Allows the definition of geocentric CS linear units for user-defined GCS.

Requirements Class: GeogLinearUnitsGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogLinearUnitsGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogLinearUnitsGeoKey.ID <i>The GeogLinearUnitsGeoKey SHALL have ID = 2052</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogLinearUnitsGeoKey.type <i>The GeogLinearUnitsGeoKey SHALL have type = SHORT</i> NOTE: Original type was DOUBLE
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogLinearUnitsGeoKey.obsolete <i>GeogLinearUnitsGeoKey values in the range 1-1023 SHALL be obsolete GeoTIFF Codes</i> NOTE: Original range was 0001-2000

Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogLinearUnitsGeoKey.linear <i>GeogLinearUnitsGeoKey values in the range 1024-32766 SHALL be EPSG Unit Of Measure (UOM) Codes that express linear units</i>
	NOTE: Original range was 9000-9099
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogLinearUnitsGeoKey.private <i>GeogLinearUnitsGeoKey values in the range 32768-65535 SHALL be private</i>

7.4.7. Requirements Class GeogLinearUnitSizeGeoKey

Allows the definition of user-defined linear geocentric units, as measured in meters.

Requirements Class: GeogLinearUnitSizeGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogLinearUnitSizeGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogLinearUnitSizeGeoKey.ID <i>The GeogLinearUnitSizeGeoKey SHALL have ID = 2053</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogLinearUnitSizeGeoKey.type <i>The GeogLinearUnitSizeGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogLinearUnitSizeGeoKey.units <i>The units of the GeogLinearUnitSizeGeoKey SHALL be meters</i>

7.4.8. Requirements Class GeogAngularUnitsGeoKey

Allows the definition of **geocentric** CS Linear units for user-defined GCS and for ellipsoids.

Requirements Class: GeogAngularUnitsGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogAngularUnitsGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogAngularUnitsGeoKey.ID <i>The GeogAngularUnitsGeoKey SHALL have ID = 2054</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogAngularUnitsGeoKey.type <i>The GeogAngularUnitsGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogAngularUnitsGeoKey.values <i>GeogAngularUnitsGeoKey values SHALL lie in the range of 9101-9108</i>
	TODO: Add a table defining the Angular Units

7.4.9. Requirements Class GeogAngularUnitSizeGeoKey

Allows the definition of user-defined angular geographic units, as measured in radians.

Requirements Class: GeogAngularUnitSizeGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogAngularUnitSizeGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogAngularUnitSizeGeoKey.ID <i>The GeogAngularUnitSizeGeoKey SHALL have ID = 2055</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogAngularUnitSizeGeoKey.type <i>The GeogAngularUnitSizeGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogAngularUnitSizeGeoKey.units <i>The units of the GeogAngularUnitSizeGeoKey SHALL be radians</i>

7.4.10. Requirements Class GeogEllipsoidGeoKey

This key may be used to specify the coded ellipsoid used in the geodetic datum of the Geographic Coordinate System.

Requirements Class: GeogEllipsoidGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogEllipsoidGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogEllipsoidGeoKey.ID <i>The GeogEllipsoidGeoKey SHALL have ID = 2056</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogEllipsoidGeoKey.type <i>The GeogEllipsoidGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogEllipsoidGeoKey.obsolete <i>GeogEllipsoidGeoKey values in the range 1-1000 SHALL be obsolete EPSG/POSC Datum Codes</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogEllipsoidGeoKey.EPSGEllipsoid <i>GeogEllipsoidGeoKey values in the range 1024-32766 SHALL be EPSG Ellipsoid Codes</i> NOTE: Original range was 7000-7999
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogEllipsoidGeoKey.user-defined <i>If the GeogEllipsoidGeoKey value is 32767 (User-Defined) then the GTCitationGeoKey SHALL be populated as well as either the GeogSemiMajorAxisGeoKey or both the GeogSemiMinorAxisGeoKey and GeogInvFlatteningGeoKey</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogEllipsoidGeoKey.private <i>GeogEllipsoidGeoKey values in the range 32768-65535 SHALL be private</i>

7.4.11. Requirements Class GeogSemiMajorAxisGeoKey

Allows the specification of user-defined Ellipsoid Semi-Major Axis (a).

Requirements Class: GeogSemiMajorAxisGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogSemiMajorAxisGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogSemiMajorAxisGeoKey.ID <i>The GeogSemiMajorAxisGeoKey SHALL have ID = 2057</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogSemiMajorAxisGeoKey.type <i>The GeogSemiMajorAxisGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogSemiMajorAxisGeoKey.units <i>The units of the GeogSemiMajorAxisGeoKey SHALL be Geocentric CS linear Units</i>

7.4.12. Requirements Class GeogSemiMinorAxisGeoKey

Allows the specification of user-defined Ellipsoid Semi-Minor Axis (b).

Requirements Class: GeogSemiMinorAxisGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogSemiMinorAxisGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogSemiMinorAxisGeoKey.ID <i>The GeogSemiMinorAxisGeoKey SHALL have ID = 2058</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogSemiMinorAxisGeoKey.type <i>The GeogSemiMinorAxisGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogSemiMinorAxisGeoKey.units <i>The units of the GeogSemiMinorAxisGeoKey SHALL be Geocentric CS linear Units</i>

7.4.13. Requirements Class GeogInvFlatteningGeoKey

Allows the specification of the **inverse** of user-defined Ellipsoid's flattening parameter (f). The eccentricity-squared e^2 of the ellipsoid is related to the non-inverted f by:

$$e^2 = 2*f - f^2$$

Note: if the ellipsoid is spherical the inverse-flattening becomes infinite; use the GeogSemiMinorAxisGeoKey instead, and set it equal to the semi-major axis length.

Requirements Class: GeogInvFlatteningGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogInvFlatteningGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogInvFlatteningGeoKey.ID <i>The GeogInvFlatteningGeoKey SHALL have ID = 2059</i>

Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogInvFlatteningGeoKey.type <i>The GeogInvFlatteningGeoKey SHALL have type = DOUBLE</i>
-------------	--

7.4.14. Requirements Class GeogAzimuthUnitsGeoKey

This key may be used to specify the angular units of measurement used to defining azimuths, in geographic coordinate systems. These may be used for defining azimuthal parameters for some projection algorithms, and may not necessarily be the same angular units used for lat-long.

Requirements Class: GeogAzimuthUnitsGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/GeogAzimuthUnitsGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogAzimuthUnitsGeoKey.ID <i>The GeogAzimuthUnitsGeoKey SHALL have ID = 2060</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogAzimuthUnitsGeoKey.type <i>The GeogAzimuthUnitsGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/GeogAzimuthUnitsGeoKey.values <i>GeogAzimuthUnitsGeoKey values SHALL lie in the range of 9101-9108</i> TODO: Add a table defining the Angular Units

7.5. Projected CS Parameter GeoKeys

The PCS range of GeoKeys includes the projection and coordinate transformation keys as well. The projection keys are included in this block since they can only be used to define projected coordinate systems.

7.5.1. Requirements Class ProjectedCSTypeGeoKey

This code is provided to specify the projected coordinate system.

Requirements Class: ProjectedCSTypeGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjectedCSTypeGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjectedCSTypeGeoKey.ID <i>The ProjectedCSTypeGeoKey SHALL have ID = 3072</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjectedCSTypeGeoKey.type <i>The ProjectedCSTypeGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjectedCSTypeGeoKey.obsolete <i>ProjectedCSTypeGeoKey values in the range 1-1000 SHALL be obsolete EPSG/POSC Datum Codes</i>

Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjectedCSTypeGeoKey.EPSGProjection <i>ProjectedCSTypeGeoKey values in the range 1024-32766 SHALL be EPSG__Projected CRS Codes</i>
	NOTE: Original range was 20000-32760
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjectedCSType.user-defined <i>If the ProjectedCSTypeGeoKey value is 32767 (User-Defined) then the PCSCitationGeoKey and ProjectionGeoKey SHALL be populated</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjectedCSTypeGeoKey.private <i>ProjectedCSTypeGeoKey values in the range 32768-65535 SHALL be private</i>

7.5.2. Requirements Class PCSCitationGeoKey

As with all the "Citation" GeoKeys, this is provided to give an ASCII reference to published documentation on the Projected Coordinate System particularly if this is a "user-defined" PCS.

Requirements Class: PCSCitationGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/PCSCitationGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/PCSCitationGeoKey.ID <i>The PCSCitationGeoKey SHALL have ID = 3073</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/PCSCitationGeoKey.type <i>The PCSCitationGeoKey SHALL have type = ASCII</i>

7.6. Projection Definition GeoKeys

With the exception of the first two keys, these are mostly projection-specific parameters, and only a few will be required for any particular projection type. Projected coordinate systems automatically imply a specific projection type, as well as specific parameters for that projection, and so the keys below will only be necessary for user-defined projected coordinate systems.

7.6.1. Requirements Class ProjectionGeoKey

Allows specification of the coordinate transformation method and projection zone parameters. Note : when associated with an appropriate Geographic Coordinate System, this forms a Projected Coordinate System.

Requirements Class: ProjectionGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjectionGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjectionGeoKey.ID <i>The ProjectionGeoKey SHALL have ID = 3074</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjectionGeoKey.type <i>The ProjectionGeoKey SHALL have type = SHORT</i>

Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjectionGeoKey.obsolete <i>ProjectionGeoKey values in the range 1-1000 SHALL be obsolete EPSG/POSC Geographic Codes</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjectionGeoKey.EPSG <i>ProjectionGeoKey values in the range 20000-32760 SHALL be valid EPSG/POSC Geographic Codes</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjectionGeoKey.private <i>ProjectionGeoKey values in the range 32768-65535 SHALL be private</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ ProjectionGeoKey.user_defined <i>A ProjectionGeoKey value of 32767 indicates a user defined projection. In this case the PCSCitationGeoKey, ProjCoordTransGeoKey, and ProjLinearUnitsGeoKey SHALL be populated</i>

7.6.2. Requirements Class ProjCoordTransGeoKey

Allows specification of the coordinate transformation method used. Note: this does not include the definition of the corresponding Geographic Coordinate System to which the projected CS is related; only the transformation method is defined here.

Requirements Class: ProjCoordTransGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjCoordTransGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCoordTransGeoKey.ID <i>The ProjCoordTransGeoKey SHALL have ID = 3075</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCoordTransGeoKey.type <i>The ProjCoordTransGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ ProjCoordTransGeoKey.transform <i>ProjCoordTransGeoKey values in the range 1-16383 SHALL be valid GeoTIFF Coordinate Transformation codes</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ ProjCoordTransGeoKey.reserved <i>ProjCoordTransGeoKey values in the range 16384-32766 SHALL be reserved by GeoTIFF</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ ProjCoordTransGeoKey.private <i>ProjCoordTransGeoKey values in the range 32768-65535 SHALL be private</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ ProjCoordTransGeoKey.user_defined <i>A ProjCoordTransGeoKey value of 32767 indicates a user defined projection. In this case the PCSCitationGeoKey SHALL be populated</i>

7.6.3. Requirements Class ProjLinearUnitsGeoKey

Defines linear units used by this projection.

Requirements Class: ProjLinearUnitsGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitsGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitsGeoKey.ID <i>The ProjLinearUnitsGeoKey SHALL have ID = 3076</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitsGeoKey.type <i>The ProjLinearUnitsGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitsGeoKey.obsolete <i>ProjLinearUnitsGeoKey values in the range 1-2000 SHALL be obsolete GeoTIFF codes</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitsGeoKey.reserved <i>ProjLinearUnitsGeoKey values in the range 2001-8999 SHALL be reserved codes</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitsGeoKey.linear_units <i>ProjLinearUnitsGeoKey values in the range 9000-9099 SHALL be EPSG linear Unit codes</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitsGeoKey.angular_units <i>ProjLinearUnitsGeoKey values in the range 9100-9199 SHALL be EPSG Angular Unit codes</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitsGeoKey.private <i>ProjLinearUnitsGeoKey values in the range 32768-65535 SHALL be private</i>

7.6.4. Requirements Class ProjLinearUnitSizeGeoKey

Defines size of user-defined linear units in meters.

Requirements Class: ProjLinearUnitSizeGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitSizeGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitSizeGeoKey.ID <i>The ProjLinearUnitSizeGeoKey SHALL have ID = 3077</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitSizeGeoKey.type <i>The ProjLinearUnitSizeGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjLinearUnitSizeGeoKey.units <i>The ProjLinearUnitSizeGeoKey SHALL have units = meters</i>

7.6.5. Requirements Class ProjStdParallel1GeoKey

Latitude of primary Standard Parallel.

Requirements Class: ProjStdParallel1GeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjStdParallel1GeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjStdParallel1GeoKey.ID <i>The ProjStdParallel1GeoKey SHALL have ID = 3078</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjStdParallel1GeoKey.type <i>The ProjStdParallel1GeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjStdParallel1GeoKey.units <i>The ProjStdParallel1GeoKey SHALL have units as specified by the GeogAngularUnit Geo Key</i>

7.6.6. Requirements Class ProjStdParallel2GeoKey

Latitude of second Standard Parallel.

Requirements Class: ProjStdParallel2GeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjStdParallel2GeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjStdParallel2GeoKey.ID <i>The ProjStdParallel2GeoKey SHALL have ID = 3079</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjStdParallel2GeoKey.type <i>The ProjStdParallel2GeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjStdParallel2GeoKey.units <i>The ProjStdParallel2GeoKey SHALL have units as specified by the GeogAngularUnit GeoKey</i>

7.6.7. Requirements Class ProjNatOriginLongGeoKey

Longitude of map-projection Natural origin.

Requirements Class: ProjNatOriginLongGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjNatOriginLongGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjNatOriginLongGeoKey.ID <i>The ProjNatOriginLongGeoKey SHALL have ID = 3080</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjNatOriginLongGeoKey.type <i>The ProjNatOriginLongGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjNatOriginLongGeoKey.units <i>The ProjNatOriginLongGeoKey SHALL have units as specified by the GeogAngularUnit Geo Key</i>

7.6.8. Requirements Class ProjNatOriginLatGeoKey

Latitude of map-projection Natural origin.

Requirements Class: ProjNatOriginLatGeoKey

http://www.opengis.net/spec/GeoTIFF/0.0/ProjNatOriginLatGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjNatOriginLatGeoKey.ID <i>The ProjNatOriginLatGeoKey SHALL have ID = 3081</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjNatOriginLatGeoKey.type <i>The ProjNatOriginLatGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjNatOriginLatGeoKey.units <i>The ProjNatOriginLatGeoKey SHALL have units as specified by the GeogAngularUnit GeoKey</i>

7.6.9. Requirements Class ProjFalseEastingGeoKey

Gives the easting coordinate of the map projection Natural origin.

Requirements Class: ProjFalseEastingGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseEastingGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseEastingGeoKey.ID <i>The ProjFalseEastingGeoKey SHALL have ID = 3082</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseEastingGeoKey.type <i>The ProjFalseEastingGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseEastingGeoKey.units <i>The ProjFalseEastingGeoKey SHALL have units as specified by the ProjLinearUnit GeoKey</i>

7.6.10. Requirements Class ProjFalseNorthingGeoKey

Gives the northing coordinate of the map projection Natural origin.

Requirements Class: ProjFalseNorthingGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseNorthingGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseNorthingGeoKey.ID <i>The ProjFalseNorthingGeoKey SHALL have ID = 3083</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseNorthingGeoKey.type <i>The ProjFalseNorthingGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseNorthingGeoKey.units <i>The ProjFalseNorthingGeoKey SHALL have units as specified by the ProjLinearUnit GeoKey</i>

7.6.11. Requirements Class ProjFalseOriginLongGeoKey

Gives the longitude of the False origin.

Requirements Class: ProjFalseOriginLongGeoKey
--

http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginLongGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginLongGeoKey.ID <i>The ProjFalseOriginLongGeoKey SHALL have ID = 3084</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginLongGeoKey.type <i>The ProjFalseOriginLongGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginLongGeoKey.units <i>The ProjFalseOriginLongGeoKey SHALL have units as specified by the GeogAngularUnit GeoKey</i>

7.6.12. Requirements Class ProjFalseOriginLatGeoKey

Gives the latitude of the False origin.

Requirements Class: ProjFalseOriginLatGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginLatGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginLatGeoKey.ID <i>The ProjFalseOriginLatGeoKey SHALL have ID = 3085</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginLatGeoKey.type <i>The ProjFalseOriginLatGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginLatGeoKey.units <i>The ProjFalseOriginLatGeoKey SHALL have units as specified by the GeogAngularUnit GeoKey</i>

7.6.13. Requirements Class ProjFalseOriginEastingGeoKey

Gives the easting coordinate of the false origin. This is NOT the False Easting, which is the easting attached to the Natural origin.

Requirements Class: ProjFalseOriginEastingGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginEastingGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginEastingGeoKey.ID <i>The ProjFalseOriginEastingGeoKey SHALL have ID = 3086</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginEastingGeoKey.type <i>The ProjFalseOriginEastingGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginEastingGeoKey.units <i>The ProjFalseOriginEastingGeoKey SHALL have units as specified by the ProjLinearUnit GeoKey</i>

7.6.14. Requirements Class ProjFalseOriginNorthingGeoKey

Gives the northing coordinate of the False origin. This is NOT the False Northing, which is the northing attached to the Natural origin.

Requirements Class: ProjFalseOriginNorthingGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginNorthingGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginNorthingGeoKey.ID <i>The ProjFalseOriginNorthingGeoKey SHALL have ID = 3087</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginNorthingGeoKey.type <i>The ProjFalseOriginNorthingGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjFalseOriginNorthingGeoKey.units <i>The ProjFalseOriginNorthingGeoKey SHALL have units as specified by the ProjLinearUnit GeoKey</i>

7.6.15. Requirements Class ProjCenterLongGeoKey

Longitude of Center of Projection. Note that this is not necessarily the origin of the projection.

Requirements Class: ProjCenterLongGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterLongGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterLongGeoKey.ID <i>The ProjCenterLongGeoKey SHALL have ID = 3088</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterLongGeoKey.type <i>The ProjCenterLongGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterLongGeoKey.units <i>The ProjCenterLongGeoKey SHALL have units as specified by the GeogAngularUnit GeoKey</i>

7.6.16. Requirements Class ProjCenterLatGeoKey

Latitude of Center of Projection. Note that this is not necessarily the origin of the projection.

Requirements Class: ProjCenterLatGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterLatGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterLatGeoKey.ID <i>The ProjCenterLatGeoKey SHALL have ID = 3089</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterLatGeoKey.type <i>The ProjCenterLatGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterLatGeoKey.units <i>The ProjCenterLatGeoKey SHALL have units as specified by the GeogAngularUnit GeoKey</i>

7.6.17. Requirements Class ProjCenterEastingGeoKey

Gives the easting coordinate of the center. This is NOT the False Easting.

Requirements Class: ProjCenterEastingGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterEastingGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterEastingGeoKey.ID <i>The ProjCenterEastingGeoKey SHALL have ID = 3090</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterEastingGeoKey.type <i>The ProjCenterEastingGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterEastingGeoKey.units <i>The ProjCenterEastingGeoKey SHALL have units as specified by the ProjLinearUnit GeoKey</i>

7.6.18. Requirements Class ProjCenterNorthingGeoKey

Gives the northing coordinate of the center. This is NOT the False Northing.

Requirements Class: ProjCenterNorthingGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterNorthingGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterNorthingGeoKey.ID <i>The ProjCenterNorthingGeoKey SHALL have ID = 3091</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterNorthingGeoKey.type <i>The ProjCenterNorthingGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjCenterNorthingGeoKey.units <i>The ProjCenterNorthingGeoKey SHALL have units as specified by the ProjLinearUnit GeoKey</i>

7.6.19. Requirements Class ProjScaleAtNatOriginGeoKey

Scale at Natural Origin. This is a ratio, so no units are required.

Requirements Class: ProjScaleAtNatOriginGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjScaleAtNatOriginGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjScaleAtNatOriginGeoKey.ID <i>The ProjScaleAtNatOriginGeoKey SHALL have ID = 3092</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjScaleAtNatOriginGeoKey.type <i>The ProjScaleAtNatOriginGeoKey SHALL have type = DOUBLE</i>

7.6.20. Requirements Class ProjScaleAtCenterGeoKey

Scale at Center. This is a ratio, so no units are required.

Requirements Class: ProjScaleAtCenterGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjScaleAtCenterGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjScaleAtCenterGeoKey.ID <i>The ProjScaleAtCenterGeoKey SHALL have ID = 3093</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjScaleAtCenterGeoKey.type <i>The ProjScaleAtCenterGeoKey SHALL have type = DOUBLE</i>

7.6.21. Requirements Class ProjAzimuthAngleGeoKey

Azimuth angle east of true north of the central line passing through the projection center (for elliptical (Hotine) Oblique Mercator). Note that this is the standard method of measuring azimuth, but is opposite the usual mathematical convention of positive indicating counter-clockwise.

Requirements Class: ProjAzimuthAngleGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjAzimuthAngleGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjAzimuthAngleGeoKey.ID <i>The ProjAzimuthAngleGeoKey SHALL have ID = 3094</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjAzimuthAngleGeoKey.type <i>The ProjAzimuthAngleGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjAzimuthAngleGeoKey.units <i>The ProjAzimuthAngleGeoKey SHALL have units as specified by the GeogAzimuthUnit GeoKey</i>

7.6.22. Requirements Class ProjStraightVertPoleLongGeoKey

Longitude at Straight Vertical Pole. For polar stereographic.

Requirements Class: ProjStraightVertPoleLongGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/ProjStraightVertPoleLongGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjStraightVertPoleLongGeoKey.ID <i>The ProjStraightVertPoleLongGeoKey SHALL have ID = 3095</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjStraightVertPoleLongGeoKey.type <i>The ProjStraightVertPoleLongGeoKey SHALL have type = DOUBLE</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/ProjStraightVertPoleLongGeoKey.units <i>The ProjStraightVertPoleLongGeoKey SHALL have units as specified by the GeogAngularUnit GeoKey</i>

7.7. Vertical CS Parameter Keys

Note: Vertical coordinate systems are not yet implemented. These sections are provided for future development, and any vertical coordinate systems in the current revision must be defined using the VerticalCitationGeoKey.

7.7.1. Requirements Class VerticalCSTypeGeoKey

This key may be used to specify the vertical coordinate system.

Requirements Class: VerticalCSTypeGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/VerticalCSTypeGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalCSTypeGeoKey.ID <i>The VerticalCSTypeGeoKey SHALL have ID = 4096</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalCSTypeGeoKey.type <i>The VerticalCSTypeGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalCSTypeGeoKey.reserved <i>VerticalCSTypeGeoKey values in the range 1-1023 SHALL be reserved</i> NOTE: Original ranges were 0001-4999 and 6000-32766
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalCSTypeGeoKey.EPSGValues <i>VerticalCSTypeGeoKey values in the range 1024-32766 SHALL be EPSG Vertical CRS Codes</i> NOTE: Original ranges were 5000-5099 and 5200-5999
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalCSTypeGeoKey.private <i>VerticalCSTypeGeoKey values in the range 32768-65535 SHALL be private</i>

7.7.2. Requirements Class VerticalCitationGeoKey

This key may be used to document the vertical coordinate system used, and its parameters.

Requirements Class: VerticalCitationGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/VerticalCitationGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalCitationGeoKey.ID <i>The VerticalCitationGeoKey SHALL have ID = 4097</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalCitationGeoKey.type <i>The VerticalCitationGeoKey SHALL have type = ASCII</i>

7.7.3. Requirements Class VerticalDatumGeoKey

This key may be used to specify the vertical datum for the vertical coordinate system.

Requirements Class: VerticalDatumGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/VerticalDatumGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalDatumGeoKey.ID <i>The VerticalDatumGeoKey SHALL have ID = 4098</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalDatumGeoKey.type <i>The VerticalDatumGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalDatumGeoKey.VertDatum <i>VerticalDatumGeoKey values in the range 1-16383 SHALL be Vertical Datum Codes</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalDatumGeoKey.reserved <i>VerticalDatumGeoKey values in the range 16384-32766 SHALL be reserved</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalDatumGeoKey.private <i>VerticalDatumGeoKey values in the range 32768-65535 SHALL be private</i>

7.7.4. Requirements Class VerticalUnitsGeoKey

This key may be used to specify the vertical units of measurement used in the geographic coordinate system, in cases where geographic CS's need to reference the vertical coordinate. This, together with the Citation key, comprise the only fully implemented keys in this section, at present.

Requirements Class: VerticalUnitsGeoKey	
http://www.opengis.net/spec/GeoTIFF/0.0/VerticalUnitsGeoKey	
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalUnitsGeoKey.ID <i>The VerticalUnitsGeoKey SHALL have ID = 4099</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalUnitsGeoKey.type <i>The VerticalUnitsGeoKey SHALL have type = SHORT</i>
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalUnitsGeoKey.obsolete <i>VerticalUnitsGeoKey values in the range 1-1023 SHALL be obsolete GeoTIFF Codes</i> NOTE: Original range was 0001-2000 for obsolete codes and 2001-8999 for reserved
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalUnitsGeoKey.linear <i>VerticalUnitsGeoKey values in the range 1024-32766 SHALL be EPSG Unit of Measure (UOM) Codes that express linear units</i> NOTE: Original ranges were 9000-9099 and 9100-9199
Requirement	http://www.opengis.net/spec/GeoTIFF/0.0/VerticalUnitsGeoKey.private <i>VerticalUnitsGeoKey values in the range 32768-65535 SHALL be private</i>

Chapter 8. Media Types for any data encoding(s)

A section describing the MIME-types to be used is mandatory for any standard involving data encodings. If no suitable MIME type exists in <http://www.iana.org/assignments/media-types/index.html> then this section may be used to define a new MIME type for registration with IANA.

Annex A: Conformance Class Abstract Test Suite (Normative)

NOTE

Ensure that there is a conformance class for each requirements class and a test for each requirement (identified by requirement name and number)

A.1. Conformance Class A

A.1.1. Requirement 1

Test id:	/conf/conf-class-a/req-name-1
Requirement:	/req/req-class-a/req-name-1
Test purpose:	Verify that...
Test method:	Inspect...

A.1.2. Requirement 2

Annex B: The GeoTIFF File Structure (Informative)

B.1. Introduction

The current GeoTIFF specification (Ritter and Ruth, 1995) includes a detailed description of the structural approach used in GeoTIFF and the semantics and values of the tags. The tag specifications are included in Clause 7 of this standard as requirements. This Annex provides an informative overview of the structure of a GeoTIFF file and tags. Much of this information is excerpted from Ritter and Ruth, 1995.

GeoTIFF fully complies with the TIFF 6.0 specifications, and its extensions do not in any way go against the TIFF recommendations, nor do they limit the scope of raster data supported by TIFF.

GeoTIFF uses a small set of reserved TIFF tags to store a broad range of georeferencing information, catering to geographic as well as projected coordinate systems needs. Projections include UTM, US State Plane and National Grids, as well as the underlying projection types such as Transverse Mercator, Lambert Conformal Conic, etc. No information is stored in private structures, IFD's or other mechanisms that would hide information from naive TIFF reading software.

GeoTIFF uses a "MetaTag" (GeoKey) approach to encode dozens of information elements into just 6 tags, taking advantage of TIFF platform-independent data format representation to avoid cross-platform interchange difficulties. These keys are designed in a manner parallel to standard TIFF tags, and closely follow the TIFF discipline in their structure and layout. New keys may be defined as needs arise, within the current framework, and without requiring the allocation of new tags from Aldus/Adobe.

GeoTIFF uses numerical codes to describe projection types, coordinate systems, datums, ellipsoids, etc. The projection, datums and ellipsoid codes are derived from the EPSG list compiled by the Petrotechnical Open Software Corporation (POSC), and mechanisms for adding further international projections, datums and ellipsoids has been established. The GeoTIFF information content is designed to be compatible with the data decomposition approach used by the National Spatial Data Infrastructure (NSDI) of the U.S. Federal Geographic Data Committee (FGDC).

While GeoTIFF provides a robust framework for specifying a broad class of existing Projected coordinate systems, it is also fully extensible, permitting internal, private or proprietary information storage. However, since this standard arose from the need to avoid multiple proprietary encoding systems, use of private implementations is to be discouraged.

B.2. GeoTIFF Design Considerations

Every effort has been made to adhere to the philosophy of TIFF data abstraction. The GeoTIFF tags conform to a hierarchical data structure of tags and keys, similar to the tags which have been implemented in the "basic" and "extended" TIFF tags already supported in TIFF Version 6 specification. The following are some points considered in the design of GeoTIFF:

- Private binary structures, while permitted under the TIFF spec, are in general difficult to

maintain, and are intrinsically platform- dependent. Whenever possible, information should be sorted into their intrinsic data-types, and placed into appropriately named tags. Also, implementers of TIFF readers would be more willing to honor a new tag specification if it does not require parsing novel binary structures.

- Any Tag value which is to be used as a "keyword" switch or modifier mistakes of mis-spelling a keyword, as well as facilitating an implementation in code using the "switch/case" features of most languages. In general, scanning ASCII strings for keywords (CaseINSensitive?) is a hazardous (not to mention slower and more complex) operation.
- True "Extensibility" strongly suggests that the Tags defined have a sufficiently abstract definition so that the same tag and its values may be used and interpreted in different ways as more complex information spaces are developed. For example, the old SubFileType tag (255) had to be obsoleted and replaced with a NewSubFileType tag, because images began appearing which could not fit into the narrowly defined classes for that Tag. Conversely, the YCbCrSubsampling Tag has taken on new meaning and importance as the JPEG compression standard for TIFF becomes finalized.

B.3. GeoTIFF Software Requirements

GeoTIFF requires support for all documented TIFF 6.0 tag data-types, and in particular requires the IEEE double-precision floating-point "DOUBLE" type tag. Most of the parameters for georeferencing will not have sufficient accuracy with single-precision IEEE, nor with RATIONAL format storage. The only other alternative for storing high-precision values would be to encode as ASCII, but this does not conform to TIFF recommendations for data encoding.

It is worth emphasizing here that the TIFF spec indicates that TIFF-compliant readers shall honor the 'byte-order' indicator, meaning that 4-byte integers from files created on opposite order machines will be swapped in software, and that 8-byte DOUBLE's will be 8-byte swapped.

A GeoTIFF reader/writer, in addition to supporting the standard TIFF tag types, must also have an additional module which can parse the "Geokey" MetaTag information. A public-domain software package for performing this function is now available; see the "References" (section 5 of Ritter and Ruth, 1995) for the location.

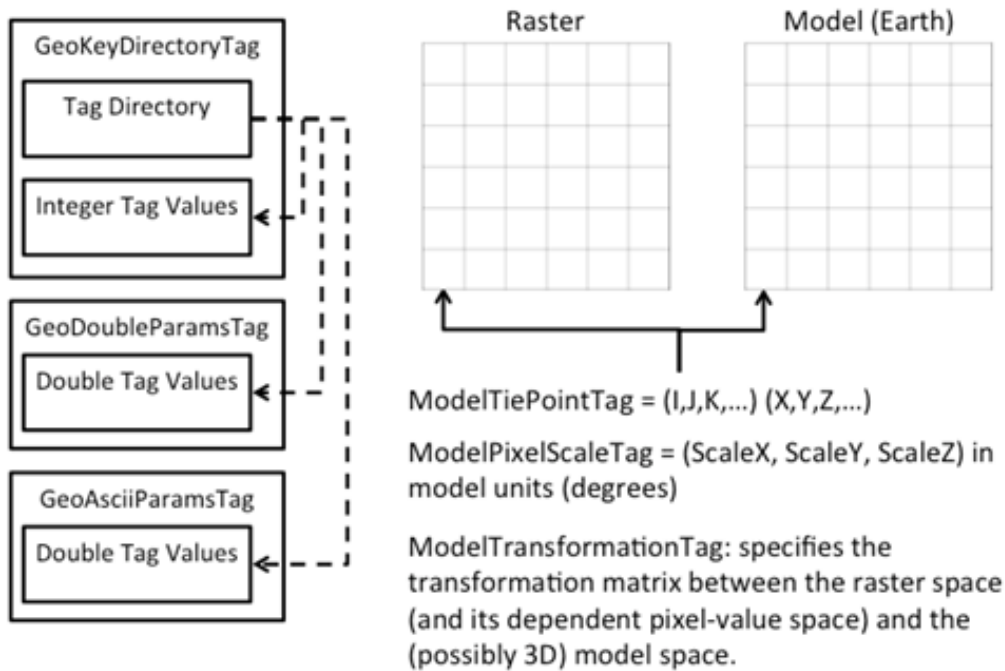
B.4. GeoTIFF File and "Key" Structure

This section describes the abstract file-format and "GeoKey" data storage mechanism used in GeoTIFF. Uses of this mechanism for implementing georeferencing and geocoding is detailed in section 2.6 and section 2.7 of Ritter and Ruth, 1995.

A GeoTIFF file is a TIFF 6.0 file, and inherits the file structure as described in the corresponding portion of the TIFF spec. All GeoTIFF specific information is encoded in several additional reserved TIFF tags, and contains no private Image File Directories (IFD's), binary structures or other private information invisible to standard TIFF readers.

The number and type of parameters that would be required to describe most popular projection types would, if implemented as separate TIFF tags, likely require dozens or even hundred of tags, exhausting the limited resources of the TIFF tag-space. On the other hand, a private IFD, while

providing thousands of free tags, is limited in that its tag-values are invisible to non-savvy TIFF readers (which don't know that the IFD_OFFSET tag value points to a private IFD).



To avoid these problems, a GeoTIFF file stores projection parameters in a set of "Keys" which are virtually identical in function to a "Tag", but has one more level of abstraction above TIFF. Effectively, it is a sort of "Meta-Tag". A Key works with formatted tag-values of a TIFF file the way that a TIFF file deals with the raw bytes of a data file. Like a tag, a Key has an ID number ranging from 0 to 65535, but unlike TIFF tags, all key ID's are available for use in GeoTIFF parameter definitions.

The Keys in GeoTIFF (also call "GeoKeys") are all referenced from the GeoKeyDirectoryTag, which defined as follows (Clause 7.1):

```
GeoKeyDirectoryTag:
  Tag = 34735 (87AF.H)
  Type = SHORT (2-byte unsigned short)
  N = variable, >= 4
  Alias: ProjectionInfoTag, CoordSystemInfoTag
  Owner: SPOT Image, Inc.
```

This tag may be used to store the GeoKey Directory, which defines and references the "GeoKeys", as described below.

The tag is an array of unsigned SHORT values, which are primarily grouped into blocks of 4. The first 4 values are special, and contain GeoKey directory header information. The header values consist of the following information, in order:

Header={KeyDirectoryVersion, KeyRevision, MinorRevision, NumberOfKeys} where

- *KeyDirectoryVersion* indicates the current version of Key implementation, and will only change if this Tag's Key structure is changed. (Similar to the TIFFVersion (42)). The current

DirectoryVersion number is 1. This value will most likely never change, and may be used to ensure that this is a valid Key-implementation.

- *KeyRevision* indicates what revision of Key-Sets are used.
- *MinorRevision* indicates what set of Key-codes are used. The complete revision number is denoted <KeyRevision>.<MinorRevision>
- *NumberOfKeys* indicates how many Keys are defined by the rest of this Tag.

This header is immediately followed by a collection of <NumberOfKeys> KeyEntry sets, each of which is also 4-SHORTS long. Each KeyEntry is modeled on the "TIFFEntry" format of the TIFF directory header, and is of the form:

- KeyEntry = { KeyID, TIFFTagLocation, Count, Value_Offset } where
- *KeyID* gives the key-ID value of the Key (identical in function to TIFF tag ID, but completely independent of TIFF tag-space),
- *TIFFTagLocation* indicates which TIFF tag contains the value(s) of the Key: if TIFFTagLocation is 0, then the value is SHORT, and is contained in the "Value_Offset" entry. Otherwise, the type (format) of the value is implied by the TIFF-Type of the tag containing the value.
- *Count* indicates the number of values in this key.
- *Value_Offset* Value_Offset indicates the index-offset **into** the TagArray indicated by TIFFTagLocation, if it is nonzero. If TIFFTagLocation=0, then Value_Offset contains the actual (SHORT) value of the Key, and Count=1 is implied. **Note that the offset is not a byte-offset, but rather an index based on the natural data type of the specified tag array.**

Following the KeyEntry definitions, the KeyDirectory tag may also contain additional values. For example, if a Key requires multiple SHORT values, they shall be placed at the end of this tag, and the KeyEntry will set TIFFTagLocation=GeoKeyDirectoryTag, with the Value_Offset pointing to the location of the value(s).

All key-values which are not of type SHORT are to be stored in one of the following two tags, based on their format:

```
GeoDoubleParamsTag:  
  Tag = 34736 (87B0.H)  
  Type = DOUBLE (IEEE Double precision)  
  N = variable  
  Owner: SPOT Image, Inc.
```

This tag is used to store all of the DOUBLE valued GeoKeys, referenced by the GeoKeyDirectoryTag. The meaning of any value of this double array is determined from the GeoKeyDirectoryTag reference pointing to it. FLOAT values should first be converted to DOUBLE and stored here.

```
GeoAsciiParamsTag:
  Tag = 34737 (87B1.H)
  Type = ASCII
  Owner: SPOT Image, Inc.
  N = variable
```

This tag is used to store all of the ASCII valued GeoKeys, referenced by the GeoKeyDirectoryTag. Since keys use offsets into tags, any special comments may be placed at the beginning of this tag. For the most part, the only keys that are ASCII valued are "Citation" keys, giving documentation and references for obscure projections, datums, etc.

Note on ASCII Keys:

Special handling is required for ASCII-valued keys. While it is true that TIFF 6.0 permits multiple NULL-delimited strings within a single ASCII tag, the secondary strings might not appear in the output of naive "tiffdump" programs. For this reason, the null delimiter of each ASCII Key value shall be converted to a "|" (pipe) character before being installed back into the ASCII holding tag, so that a dump of the tag will look like this.

```
AsciiTag="first_value|second_value|etc...last_value|"
```

A baseline GeoTIFF-reader must check for and convert the final "|" pipe character of a key back into a NULL before returning it to the client software.

GeoKey Sort Order:

In the TIFF spec it is required that TIFF tags be written out to the file in tag-ID sorted order. This is done to avoid forcing software to perform N-squared sort operations when reading and writing tags.

To follow the TIFF philosophy, GeoTIFF-writers shall store the GeoKey entries in key-sorted order within the GeoKeyDirectoryTag.

```
Example:
GeoKeyDirectoryTag=( 1, 1, 2, 6,
                    1024, 0, 1, 2,
                    1026, 34737,12, 0,
                    2048, 0, 1, 32767,
                    2049, 34737,14, 12,
                    2050, 0, 1, 6,
                    2051, 34736, 1, 0 )
GeoDoubleParamsTag(34736)=(1.5)
GeoAsciiParamsTag(34737)=("Custom File|My Geographic|")
```

The first line indicates that this is a Version 1 GeoTIFF GeoKey directory, the keys are Rev. 1.2, and there are 6 Keys defined in this tag.

The next line indicates that the first Key (ID=1024 = GTModelTypeGeoKey) has the value 2 (Geographic), explicitly placed in the entry list (since TIFFTagLocation=0). The next line indicates that the Key 1026 (the GTCitationGeoKey) is listed in the GeoAsciiParamsTag (34737) array, starting at offset 0 (the first in array), and running for 12 bytes and so has the value "Custom File" (the "|" is converted to a null delimiter at the end). Going further down the list, the Key 2051 (GeogLinearUnitSizeGeoKey) is located in the GeoDoubleParamsTag (34736), at offset 0 and has the value 1.5; the value of key 2049 (GeogCitationGeoKey) is "My Geographic".

The TIFF layer handles all the problems of data structure, platform independence, format types, etc, by specifying byte-offsets, byte-order format and count, while the Key describes its key values at the TIFF level by specifying Tag number, array-index, and count. Since all TIFF information occurs in TIFF arrays of some sort, we have a robust method for storing anything in a Key that would occur in a Tag.

With this Key-value approach, there are 65536 Keys which have all the flexibility of TIFF tag, with the added advantage that a TIFF dump will provide all the information that exists in the GeoTIFF implementation.

This GeoKey mechanism is used extensively in Clause 7 and section 8.2.3 of Ritter and Ruth, 1995 where the parameters for defining Coordinate Systems and their underlying projections are defined.

B.5. Coordinate Reference Systems in GeoTIFF

GeoTIFF has been designed so that standard map coordinate reference system definitions can be readily stored in a single registered TIFF tag. It has also been designed to allow the description of coordinate reference system definitions that are non-standard, and for the description of transformations between coordinate reference systems, through the use of three or four additional TIFF tags.

However, in order for the information to be correctly exchanged between various clients and providers of GeoTIFF, it is important to establish a common system for describing map projections.

In the TIFF/GeoTIFF framework, there are essentially three different spaces upon which coordinate systems may be defined. The spaces are:

1. The raster space (Image space) R, used to reference the pixel values in an image,
2. The Device space D, and
3. The Model space, M, used to reference points on the earth.

In the sections that follow we shall discuss the relevance and use of each of these spaces, and their corresponding coordinate systems, from the standpoint of GeoTIFF.

B.5.1. Device Space and GeoTIFF


In standard TIFF 6.0 there are tags that relate raster space R with device space D, such as monitor, scanner or printer. The list of such tags consists of the following:

ResolutionUnit (296)
XResolution (282)
YResolution (283)
Orientation (274)
XPosition (286)
YPosition (287)


In GeoTIFF, provision is made to identify earth-referenced coordinate systems (model space M) and to relate M space with R space. This provision is independent of and can co-exist with the relationship between raster and device spaces. To emphasize the distinction, this spec shall not refer to "X" and "Y" raster coordinates, but rather to raster space "J" (row) and "I" (column) coordinate variables instead, as defined in section 2.5.2.2.

B.5.2. Raster Coordinate Systems

Raster Data

Raster data consists of spatially coherent, digitally stored numerical data, collected from sensors, scanners, or in other ways numerically derived. The manner in which this storage is implemented in a TIFF file is described in the standard TIFF specification. 

Raster data values, as read in from a file, are organized by software into two-dimensional arrays, the indices of the arrays being used as coordinates. There may also be additional indices for multispectral data, but these indices do not refer to spatial coordinates but spectral, and so are not of concern here.

Many different types of raster data may be georeferenced, and there  may be subtle ways in which the nature of the data itself influences how the **coordinate system** (Raster Space) is defined for raster data. For example, pixel data derived from imaging devices and sensors represent aggregate values collected over a small, finite, geographic area, and so it is natural to define coordinate systems in which the pixel value is thought of as filling an area. On the other hand, digital elevations models may consist of discrete "postings", which may best be considered as point measurements at the vertices of a grid, and not in the interior of a cell.

Raster Space

The choice of origin for raster space is not entirely arbitrary, and depends upon the nature of the data collected. Raster space coordinates shall be referred to by their pixel types, i.e., as "PixelIsArea" or "PixelIsPoint".

Note: For simplicity, both raster spaces documented below use a fixed pixel size and spacing of 1. Information regarding the visual representation of this data, such as pixels with non-unit aspect ratios, scales, orientations, etc., are best communicated with the TIFF 6.0 standard tags.

"PixelIsArea" Raster Space

The "PixelIsArea" raster grid space R, which is the default, uses coordinates I and J, with (0,0) denoting the upper-left corner of the image, and increasing I to the right, increasing J down. The first pixel-value fills the square grid cell with the bounds:

top-left = (0,0), bottom-right = (1,1)

and so on; by extension this one-by-one grid cell is also referred to as a pixel. An N by M pixel image covers an area with the mathematically defined bounds (0,0), (N,M).

```
(0,0)
+---+---+--> I
| * | * |
+---+---+      Standard (PixelIsArea) TIFF Raster space R,
| (1,1) (2,1) showing the areas (*) of several pixels.
|
J
```

"PixelIsPoint" Raster Space

The PixelIsPoint raster grid space R uses the same coordinate axis names as used in PixelIsArea Raster space, with increasing I to the right, increasing J down. The first pixel-value however, is realized as a point value located at (0,0). An N by M pixel image consists of points which fill the mathematically defined bounds (0,0), (N-1,M-1).

```
(0,0) (1,0)
*-----*-----> I
|         |
|         |      PixelIsPoint TIFF Raster space R,
*-----*      showing the location (*) of several pixels.
| (1,1)
J
```

If a point-pixel image were to be displayed on a display device with pixel cells having the same size as the raster spacing, then the upper-left corner of the displayed image would be located in raster space at (-0.5, -0.5).

B.5.3. Model Coordinate Reference Systems

The following methods of describing spatial model locations (as opposed to raster) are recognized in GeoTIFF:

- Geographic coordinates
- Geocentric coordinates
- Projected coordinates
- Vertical coordinates

Geographic, geocentric and projected coordinates are all imposed on models of the earth. To describe a location uniquely, a coordinate set must be referenced to an adequately defined coordinate reference system. If a coordinate reference system is from the GeoTIFF standard

definitions, the only reference required is the standard coordinate reference system code/name. If the coordinate reference system is non-standard, it must be defined. The required definitions are described below.

Projected coordinates, local grid coordinates, and geographical 2D coordinates, form two-dimensional horizontal coordinate systems (i.e., horizontal with respect to the earth's surface). Height is not part of these systems. To describe a position in three dimensions it is necessary to consider height as a second one-dimensional vertical coordinate reference system.

To georeference an image in GeoTIFF, you must specify a Raster Space coordinate system, choose a horizontal model coordinate system, and a transformation between these two, as will be described in section 2.6.

Geographic Coordinate Reference Systems

Geographic Coordinate Reference Systems are those that relate angular latitude and longitude (and optionally geodetic height) to an actual point on the earth. The process by which this is accomplished is rather complex, and so we describe the components of the process in detail here.

Ellipsoidal Models of the Earth

The geoid - the earth stripped of all topography - forms a reference surface for the earth. However, because it is related to the earth's gravity field, the geoid is a very complex surface; indeed, at a detailed level its description is not well known. The geoid is therefore not used in practical mapping.

It has been found that an oblate ellipsoid (an ellipse rotated about its minor axis) is a good approximation to the geoid and therefore a good model of the earth. Many approximations exist: several hundred ellipsoids have been defined for scientific purposes and about 30 are in day-to-day use for mapping. The size and shape of these ellipsoids can be defined through two parameters. GeoTIFF requires one of these to be

the semi-major axis (a),

and the second to be either

the inverse flattening ($1/f$)

or

the semi-minor axis (b).

Historical models exist which use a spherical approximation; such models are not recommended for modern applications, but if needed the size of a model sphere may be defined by specifying identical values for the semi-major and semi-minor axes; the inverse flattening cannot be used as it becomes infinite for perfect spheres.

Other ellipsoid parameters needed for mapping applications, for example the square of the eccentricity, can easily be calculated by an application from the two defining parameters. Note that GeoTIFF uses the modern geodesy convention for the symbol (b) for the semi-minor axis. No provision is made for mapping other planets in which a tri-dimensional (triaxial) ellipsoid might be required, where (b) would represent the semi-major axis and (c) the semi-minor axis.

Numeric codes for ellipsoids regularly used for earth mapping are included in the GeoTIFF reference lists.

Latitude and Longitude

The coordinate axes of the system referencing points on an ellipsoid are called latitude and longitude. More precisely, **geodetic** latitude and longitude are required in this GeoTIFF standard. A discussion of the several other types of latitude and longitude is beyond the scope of this document as they are not required for conventional mapping.

Latitude is defined to be the angle subtended with the ellipsoid's equatorial plane by a perpendicular through the surface of the ellipsoid from a point. Latitude is positive if north of the equator, negative if south.

Longitude is defined to be the angle measured about the minor (polar) axis of the ellipsoid from a prime meridian (see below) to the meridian through a point, positive if east of the prime meridian and negative if west. Unlike latitude, which has a natural origin at the equator, there is no feature on the ellipsoid which forms a natural origin for the measurement of longitude. The zero longitude can be any defined meridian. Historically, nations have used the meridian through their national astronomical observatories, giving rise to several prime meridians. By international convention, the meridian through Greenwich, England is the standard prime meridian. Longitude is only unambiguous if the longitude of its prime meridian relative to Greenwich is given. Prime meridians other than Greenwich that are sometimes used for earth mapping are included in the GeoTIFF reference lists.

Geodetic Datums

As well as there being several ellipsoids in use to model the earth, any one particular ellipsoid can have its location and orientation relative to the earth defined in different ways. If the relationship between the ellipsoid and the earth is changed, then the geographical coordinates of a point will change.

Conversely, for geographical coordinates to uniquely describe a location, the relationship between the earth and the ellipsoid must be defined. This relationship is described by a geodetic datum. An exact geodetic definition of geodetic datums is beyond the current scope of GeoTIFF. However the GeoTIFF standard requires that the geodetic datum being utilized be identified by numerical code. If required, defining parameters for the geodetic datum can be included as a citation.

Defining Geographic Coordinate Reference Systems

In summary, geographic coordinates are only unique if qualified by the code of the geographic coordinate reference system to which they belong. A geographic coordinate system has two axes, latitude and longitude, which are only unambiguous when both of the related prime meridian and geodetic datum are given, and in turn the geodetic datum definition includes the definition of an

ellipsoid. The GeoTIFF standard includes a list of frequently used geographic coordinate reference systems and their component ellipsoids, geodetic datums and prime meridians. Within the GeoTIFF standard a geographic coordinate reference system can be identified either by

the code of a standard geographic coordinate reference system

or by

a user-defined system.

The user is expected to provide geographic coordinate reference system code/name, geodetic datum code/name, ellipsoid code (if in standard) or ellipsoid name and two defining parameters (a) and either (1/f) or (b), and prime meridian code (if in standard) or name and longitude relative to Greenwich.

Geocentric Coordinate Systems

A geocentric coordinate system is a 3-dimensional coordinate system with its origin at or near the center of the earth and with 3 orthogonal axes. The Z-axis is in or parallel to the earth's axis of rotation (or to the axis around which the rotational axis precesses). The X-axis is in or parallel to the plane of the equator and passes through its intersection with the Greenwich meridian, and the Y-axis is in the plane of the equator forming a right-handed coordinate system with the X and Z axes.

Geocentric coordinate systems are not frequently used for describing locations, but they are often utilized as an intermediate step when transforming between geographic coordinate reference systems. (Coordinate transformations are described in section 2.6 below).

In the GeoTIFF standard, a geocentric coordinate reference system can be identified, either

through the geographic code (which in turn implies a datum),

or

through a user-defined name.

Projected Coordinate Systems

Although a geographical coordinate reference system is mathematically two-dimensional, it describes a three-dimensional object and cannot be represented on a plane surface without distortion. Map projections are transformations of geographical coordinates to plane coordinates in which the characteristics of the distortions are controlled. A map projection consists of a transformation method and a set of defining parameters. A projected coordinate reference system (PCS) is a two dimensional (horizontal) coordinate set which, for a specific map projection, has a single and unambiguous transformation to a geographic coordinate reference system.

In GeoTIFF PCS's are defined using the EPSG system, in which the PCS planar coordinate reference system, the Geographic coordinate reference system, and the transformation between them, are broken down into simpler logical components. Here are schematic formulas showing how the Projected Coordinate Reference Systems and Geographic Coordinates Reference Systems are encoded:

```
Projected_CS = Geographic_CS + Projection
Geographic_CS = Angular_Unit + Geodetic_Datum + Prime_Meridian
Projection = Linear_Unit + Coord_Transf_Method + CT_Parameters
Coord_Transf_Method = \{ TransverseMercator | LambertCC | ...}
CT_Parameters = \{OriginLatitude + StandardParallel+...}
```

(See also the Reference Parameters documentation in section 2.5.4).

Notice that "Transverse Mercator" is not referred to as a "Projection", but rather as a "Coordinate Transformation Method"; in GeoTIFF, as in EPSG, the word "Projection" is reserved for particular, well-defined systems in which both the coordinate transformation method, its defining parameters, and their linear units are established.

Several tens of coordinate transformation methods have been developed. Many are very similar and for practical purposes can be considered to give identical results. For example in the GeoTIFF standard Gauss-Kruger and Gauss-Boaga projection types are considered to be of the type Transverse Mercator. GeoTIFF includes a listing of commonly used projection defining parameters.

Different algorithms require different defining parameters. A future version of GeoTIFF will include formulas for specific map projection algorithms recommended for use with listed projection parameters.

To limit the magnitude of distortions of projected coordinate systems, the boundaries of usage are sometimes restricted. To cover more extensive areas, two or more projected coordinate systems may be required. In some cases many of the defining parameters of a set of projected coordinate reference systems will be held constant.

The GeoTIFF standard does not impose a strict hierarchy onto such zoned systems such as US State Plane or UTM, but considers each zone to be a discrete projected coordinate reference system; the ProjectedCSTypeGeoKey code value alone is sufficient to identify the standard coordinate reference systems.

Within the GeoTIFF standard a projected coordinate reference system can be identified either by

the code of a standard projected coordinate reference system

or by

a user-defined system.

User-define projected coordinate reference systems may be defined by defining the Geographic

Coordinate Reference System, the coordinate transformation method and its associated parameters, as well as the planar system's linear units.

Vertical Coordinate Reference Systems_

Many uses of GeoTIFF will be limited to a two-dimensional, horizontal, description of location for which geographic 2D coordinate reference systems and projected coordinate reference systems are adequate. If a three-dimensional description of location is required, GeoTIFF allows this either through the use of a geographic 3D coordinate reference system, a geocentric coordinate reference system or by defining a vertical coordinate reference system and using this together with a geographic 2D or projected coordinate system.

In general usage, elevations and depths are referenced to a surface at or close to the geoid. Through increasing use of satellite positioning systems the ellipsoid is increasingly being used as a vertical reference surface. The relationship between the geoid and an ellipsoid is in general not well known, but is required when coordinate reference system transformations are to be executed.

B.5.4. Reference Parameters

Most of the numerical coding systems and coordinate reference system definitions are based on the hierarchical system developed for the EPSG Geodetic Parameter Dataset ('EPSG Dataset'). The complete set of EPSG definitions is available at www.epsg-registry.org.

The EPSG Dataset is maintained by the Geodesy subcommittee of the International Association of Oil and Gas Producers (IOGP). It follows the ISO 19111 data model for describing the definitions of coordinate reference systems (CRSs) and definitions of coordinate operations (transformations, conversions, point motion operations and concatenated operations). CRSs and coordinate operations are composed of a number of entities and attributes. Some of these entities themselves are composed entities and attributes, in a nested structure.

Within the EPSG Dataset each entity has a code in the range 1024 through 32766. In the early days of the Dataset, this code range was divided into ranges, with one range for each entity type. At that time, entity codes were unique. The GeoTIFF v1.0 specification was written at this time, and the EPSG code ranges for entity types were written into the specification.

However as the number of items in the EPSG Dataset grew, some of the entity code ranges became fully assigned. The unique code system broke down. Since 2006, all entity types have been separately assigned codes within the range 1024 through 32766. Within each entity type codes remain unique, but one code may be used for several entity types. For example, code 4326 is used for both a CRS and for a geographic extent (in EPSG called 'area'). Codes at and just above the lower end of the range 1024 through 32766 may be used by numerous entity types: for example by the year 2018 code 1026 has been assigned to 10 different entity types. EPSG codes therefore are only unique when the entity type is disclosed. `EPSG::4326` is ambiguous, `crs:EPSG::4326` and `area:EPSG::4326` are unambiguous.

Since 1999 (from EPSG Dataset v5.0 and later) EPSG policy has been to never remove any invalid data but instead to leave it in the Dataset with its status set to deprecated. Deprecated data contains an error and is invalid. As such reference to the version of the EPSG Dataset is unnecessary. Using EPSG Dataset versions 5.0 and 9.3 as examples, `crs:EPSG:5.0:4326` and `crs:EPSG:9.3:4326` and

crs:EPSG::4326 reference the same object.

The 1995 GeoTIFFv1.0 specification explicitly referenced EPSG codes that were available at that time. Since then many new entries have been made available, and some available in 1995 have been withdrawn and replaced. This document removes the 1995 GeoTIFF v1.0 reference to specific EPSG codes and replaces that with allowing reference to any code in the EPSG Dataset. Reference to a coordinate reference system EPSG code is sufficient for a complete definition: it implies use of the components such as datum that are associated with that CRS in the EPSG Dataset definition.

The EPSG codes for coordinate reference system components (datum, ellipsoid, map projection, etc.) should only be referenced when describing a user-defined coordinate reference system.

The terms of use of the EPSG Dataset are given at <http://www.epsg.org/Termsfuse.aspx>.

B.6. Coordinate Transformations

The purpose of GeoTIFF is to allow the definitive identification of georeferenced locations within a raster dataset. This is generally accomplished through tying raster space coordinates to a model space coordinate system, when no further information is required. In the GeoTIFF nomenclature, "georeferencing" refers to tying raster space to a model space M, while "geocoding" refers to defining how the model space M assigns coordinates to points on the earth.

The three tags defined below may be used for defining the relationship between R and M, and the relationship may be diagrammed as:

```
ModelPixelScaleTag
ModelTiepointTag
R ----- OR -----> M
(I,J,K) ModelTransformationTag (X,Y,Z)
```

The next section describes these Baseline georeferencing tags in detail.

B.6.1. GeoTIFF Tags for Coordinate Transformations

For most common applications, the transformation between raster and model space may be defined with a set of raster-to-model tiepoints and scaling parameters. The following two tags may be used for this purpose:

```
ModelTiepointTag:
Tag = 33922 (8482.H)
Type = DOUBLE (IEEE Double precision)
N = 6*K, K = number of tiepoints
Alias: GeoreferenceTag
Owner: Intergraph
```

This tag stores raster → model tiepoint pairs in the order

```
ModelTiepointTag = (... ,I,J,K, X,Y,Z...),
```

where (I,J,K) is the point at location (I,J) in raster space with pixel-value K, and (X,Y,Z) is a vector in model space. In most cases the model space is only two-dimensional, in which case both K and Z should be set to zero; this third dimension is provided in anticipation of future support for 3D digital elevation models and vertical coordinate systems.

A raster image may be georeferenced simply by specifying its location, size and orientation in the model coordinate space M. This may be done by specifying the location of three of the four bounding corner points. However, tiepoints are only to be considered exact at the points specified; thus defining such a set of bounding tiepoints does **not** imply that the model space locations of the interior of the image may be exactly computed by a linear interpolation of these tiepoints.

However, since the relationship between the Raster space and the model space will often be an exact, affine transformation, this relationship can be defined using one set of tiepoints and the "ModelPixelScaleTag", described below, which gives the vertical and horizontal raster grid cell size, specified in model units.

If possible, the first tiepoint placed in this tag shall be the one establishing the location of the point (0,0) in raster space. However, if this is not possible (for example, if (0,0) is goes to a part of model space in which the projection is ill-defined), then there is no particular order in which the tiepoints need be listed.

For orthorectification or mosaicking applications a large number of tiepoints may be specified on a mesh over the raster image. However, the definition of associated grid interpolation methods is not in the scope of the current GeoTIFF spec.

Remark: As mentioned in section 2.5.1, all GeoTIFF information is independent of the XPosition, YPosition, and Orientation tags of the standard TIFF 6.0 spec.

The next two tags are optional tags provided for defining exact affine transformations between raster and model space; baseline GeoTIFF files may use either, but shall never use both within the same TIFF image directory.

```
ModelPixelScaleTag:  
  Tag = 33550  
  Type = DOUBLE (IEEE Double precision)  
  N = 3  
  Owner: SoftDesk
```

This tag may be used to specify the size of raster pixel spacing in the model space units, when the raster space can be embedded in the model space coordinate system without rotation, and consists of the following 3 values:

```
ModelPixelScaleTag = (ScaleX, ScaleY, ScaleZ)
```

where ScaleX and ScaleY give the horizontal and vertical spacing of raster pixels. The ScaleZ is primarily used to map the pixel value of a digital elevation model into the correct Z-scale, and so for most other purposes this value should be zero (since most model spaces are 2-D, with Z=0).

A single tiepoint in the ModelTiepointTag, together with this tag, completely determine the relationship between raster and model space; thus they comprise the two tags which Baseline GeoTIFF files most often will use to place a raster image into a "standard position" in model space.

Like the Tiepoint tag, this tag information is independent of the XPosition, YPosition, Resolution and Orientation tags of the standard TIFF 6.0 spec. However, simple reversals of orientation between raster and model space (e.g. horizontal or vertical flips) may be indicated by reversal of sign in the corresponding component of the ModelPixelScaleTag. GeoTIFF compliant readers must honor this sign-reversal convention.

This tag must not be used if the raster image requires rotation or shearing to place it into the standard model space. In such cases the transformation shall be defined with the more general ModelTransformationTag, defined below.

```
ModelTransformationTag
  Tag = 34264 (85D8.H)
  Type = DOUBLE
  N = 16
  Owner: JPL Cartographic Applications Group
```

This tag may be used to specify the transformation matrix between the raster space (and its dependent pixel-value space) and the (possibly 3D) model space. If specified, the tag shall have the following organization:

```
ModelTransformationTag = (a,b,c,d,e...m,n,o,p).
```

where

model		image
coords =	matrix *	coords
-	-	-
X	a b c d	I
Y	e f g h	J
Z	i j k l	K
1	m n o p	1
-	-	-

By convention, and without loss of generality, the following parameters are currently hard-coded and will always be the same (but must be specified nonetheless):

$$m = n = o = 0, p = 1.$$

For Baseline GeoTIFF, the model space is always 2-D, and so the matrix will have the more limited form:

$$\begin{array}{|c|c|} \hline - & - \\ \hline X & a \ b \ 0 \ d \\ \hline - & - \\ \hline Y & e \ f \ 0 \ h \\ \hline - & - \\ \hline Z & 0 \ 0 \ 0 \ 0 \\ \hline - & - \\ \hline 1 & 0 \ 0 \ 0 \ 1 \\ \hline - & - \\ \hline \end{array} = \begin{array}{|c|c|} \hline - & - \\ \hline I & \\ \hline - & - \\ \hline J & \\ \hline - & - \\ \hline K & \\ \hline - & - \\ \hline 1 & \\ \hline - & - \\ \hline \end{array}$$

Values "d" and "h" will often be used to represent translations in X and Y, and so will not necessarily be zero. All 16 values should be specified, in all cases. Only the raster-to-model transformation is defined; if the inverse transformation is required it must be computed by the client, to the desired accuracy.

This matrix tag should not be used if the ModelTiepointTag and the ModelPixelScaleTag are already defined. If only a single tiepoint (I,J,K,X,Y,Z) is specified, and the ModelPixelScale = (Sx, Sy, Sz) is specified, then the corresponding transformation matrix may be computed from them as:

$$\begin{array}{|c|c|} \hline - & - \\ \hline Sx \ 0.0 \ 0.0 \ Tx & Tx = X - I/Sx \\ \hline 0.0 \ -Sy \ 0.0 \ Ty & Ty = Y + J/Sy \\ \hline 0.0 \ 0.0 \ Sz \ Tz & Tz = Z - K/Sz \ (if \ not \ 0) \\ \hline 0.0 \ 0.0 \ 0.0 \ 1.0 \\ \hline - & - \\ \hline \end{array}$$

where the -Sy is due the reversal of direction from J increasing- down in raster space to Y increasing-up in model space.

Like the Tiepoint tag, this tag information is independent of the XPosition, YPosition, and Orientation tags of the standard TIFF 6.0 spec.

Note: In Revision 0.2 and earlier, another tag was used for this matrix, which has been renamed as follows:

```

IntergraphMatrixTag
  Tag = 33920 (8480.H)
  Type = DOUBLE
  N = 17 (Intergraph implementation) or 16 (GeoTIFF 0.2 impl.)
  Owner: Intergraph

```

This tag conflicts with an internal software implementation at Intergraph, and so its use is no longer encouraged. A GeoTIFF reader should look first for the new tag, and only if it is not found should it check for this older tag. If found, it should only consider it to contain valid GeoTIFF matrix information if the tag-count is 16; the Intergraph version uses 17 values.

B.6.2. Coordinate Transformation Data Flow

TBD

B.6.3. Cookbook for Defining Transformations

Here is a 4-step guide to producing a set of Baseline GeoTIFF tags for defining coordinate transformation information of a raster dataset.

Step 1: Establish the Raster Space coordinate system used: `RasterPixelIsArea` or `RasterPixelIsPoint`.

Step 2: Establish/define the model space Type in which the image is to be georeferenced. Usually this will be a projected coordinate Reference system (PCS). If you are geocoding this data set, then the model space is defined to be the corresponding geographic, geocentric or projected coordinate reference system (skip to the "Cookbook" section 2.7.3 first to determine this).

Step 3: Identify the nature of the transformations needed to tie the raster data down to the model space coordinate reference system:

Case 1: The model-location of a raster point (x,y) is known, but not the scale or orientations: Use the `ModelTiepointTag` to define the (X,Y,Z) coordinates of the known raster point.

Case 2: The location of three non-collinear raster points are known exactly, but the linearity of the transformation is not known.

Use the `ModelTiepointTag` to define the (X,Y,Z) coordinates of all three known raster points. Do not compute or define the `ModelPixelScale` or `ModelTransformation` tag.

Case 3: The position and scale of the data is known exactly, and no rotation or shearing is needed to fit into the model space.

Use the ModelTiepointTag to define the (X,Y,Z) coordinates of the known raster point, and the ModelPixelScaleTag to specify the scale.

Case 4: The raster data requires rotation and/or lateral shearing to fit into the defined model space:

Use the ModelTransformation matrix to define the transformation.

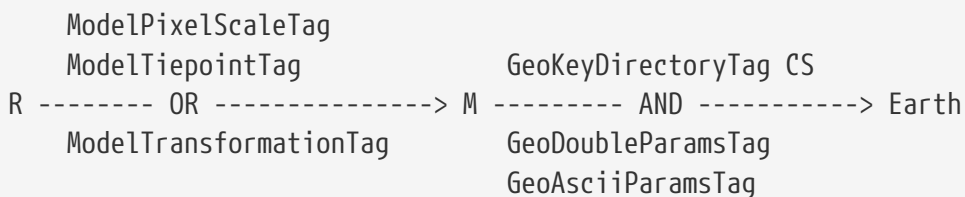
Case 5: The raster data cannot be fit into the model space with a simple affine transformation (rubber-sheeting required). Use only the ModelTiepoint tag, and specify as many tiepoints as your application requires. Note, however, that this is not a Baseline GeoTIFF implementation, and should not be used for interchange; it is recommended that the image be geometrically rectified first, and put into a standard projected coordinate reference system.

Step 4: Install the defined tag values in the TIFF file and close it.

B.7. Geocoding Raster Data

B.7.1. General Approach

A geocoded image is a georeferenced image as described in section 2.6, which also specifies a model space coordinate reference system (CS) between the model space M (to which the raster space has been tied) and the earth. The relationship can be diagrammed, including the associated TIFF tags, as follows:



The geocoding coordinate system is defined by the GeoKeyDirectoryTag, while the Georeferencing information (T) is defined by the ModelTiepointTag and the ModelPixelScale, or ModelTransformationTag. Since these two systems are independent of each other, the tags used to store the parameters are separated from each other in the GeoTIFF file to emphasize the orthogonality.

B.8. Examples

Examples of how GeoTIFF may be implemented at the Tag and GeoKey level, following the general

"Cookbook" approach above.

B.8.1. Common Examples

UTM Projected Aerial Photo

We have an aerial photo which has been orthorectified and resampled to a UTM grid, zone 60, using WGS 84 coordinate reference system; the coordinates of the upper-left corner of the image is given in easting/northing, as 350807.4m, 5316081.3m. The scanned map pixel scale is 100 meters/pixels (the actual dpi scanning ratio is irrelevant).

```
ModelTiepointTag = (0, 0, 0, 350807.4, 5316081.3, 0.0)
ModelPixelScaleTag = (100.0, 100.0, 0.0)
GeoKeyDirectoryTag:
  GTModelTypeGeoKey = 1 (ModelTypeProjected)
  GTRasterTypeGeoKey = 1 (RasterPixelIsArea)
  ProjectedCSTypeGeoKey = 32660 (PCS_WGS84_UTM_zone_60N)
  PCSCitationGeoKey = "UTM Zone 60 N with WGS84"
```

Notes:

1. We did not need to specify the GCS lat-long, since the PCS_WGS84_UTM_zone_60N codes implies particular GCS and units already (WGS_84 and meters). The citation was added just for documentation.
2. The "GeoKeyDirectoryTag" is expressed using the "GeoKey" structure defined above. At the TIFF level the tags look like this:

```
GeoKeyDirectoryTag=( 1,      0,  2,      4,
                    1024,    0,  1,      1,
                    1025,    0,  1,      1,
                    3072,    0,  1, 32660,
                    3073, 34737, 25,      0 )
GeoAsciiParamsTag(34737)=("UTM Zone 60 N with WGS84|")
```

For the rest of these examples we will only show the GeoKey-level dump, with the understanding that the actual TIFF-level tag representation can be determined from the documentation.

Standard State Plane

We have a USGS State Plane Map of Texas, Central Zone, using NAD83, correctly oriented. The map resolution is 1000 meters/pixel, at origin. There is a grid intersection line in the image at pixel location (50,100), and corresponds to the projected coordinate reference system easting/northing of (949465.0, 3070309.1).

```

ModelTiepointTag = ( 50, 100, 0, 949465.0, 3070309.1, 0)
ModelPixelScaleTag = (1000, 1000, 0)
GeoKeyDirectoryTag:
    GTModelTypeGeoKey = 1 (ModelTypeProjected)
    GTRasterTypeGeoKey = 1 (RasterPixelIsArea)
    ProjectedCSTypeGeoKey = 32139 (PCS_NAD83_Texas_Central)

```

Notice that in this case, since the PCS is a standard code, we do not need to define the GCS, datum, etc, since those are implied by the PCS code. Also, since this is NAD83, meters are used rather than US Survey feet (as in NAD 27).

Lambert Conformal Conic Aeronautical Chart

We have a 500 x 500 scanned aeronautical chart of Seattle, WA, using Lambert Conformal Conic projection, correctly oriented. The central meridian is at 120 degrees west. The map resolution is 1000 meters/pixel, at origin, and uses NAD27 datum. The standard parallels of the projection are at 41d20m N and 48d40m N. The latitude of the origin is at 45 degrees North, and occurs in the image at the raster coordinates (80,100). The origin is given a false easting and northing of 200000m, 1500000m.

```

ModelTiepointTag = ( 80, 100, 0, 200000, 1500000, 0)
ModelPixelScaleTag = (1000, 1000, 0)
GeoKeyDirectoryTag:
    GTModelTypeGeoKey = 1 (ModelTypeProjected)
    GTRasterTypeGeoKey = 1 (RasterPixelIsArea)
    GeographicTypeGeoKey = 4267 (GCS_NAD27)
    ProjectedCSTypeGeoKey = 32767 (user-defined)
    ProjectionGeoKey = 32767 (user-defined)
    ProjLinearUnitsGeoKey = 9001 (Linear_Meter)
    ProjCoordTransGeoKey = 8 (CT_LambertConfConic_2SP)
        ProjStdParallel1GeoKey = 41.333
        ProjStdParallel2GeoKey = 48.666
        ProjCenterLongGeoKey = -120.0
        ProjNatOriginLatGeoKey = 45.0
        ProjFalseEastingGeoKey, = 200000.0
        ProjFalseNorthingGeoKey, = 1500000.0

```

Notice that the Tiepoint takes the false easting and northing into account when tying the raster point (50,100) to the projection origin.

DMA ADRG Raster Graphic Map

The U.S. Defense Mapping Agency produces ARC digitized raster graphics datasets by scanning maps and geometrically resampling them into an equirectangular projection, so that they may be directly indexed with WGS84 geographic coordinates. The scale for one map is 0.2 degrees per pixel horizontally, 0.1 degrees per pixel vertically. If stored in a GeoTIFF file it contains the following information:


```
ModelTiepointTag=(0.0, 0.0, 0.0, -120.0, 32.0, 0.0)
ModelPixelScale = (0.2, 0.1, 0.0)
GeoKeyDirectoryTag:
  GTModelTypeGeoKey = 2 (ModelTypeGeographic)
  GTRasterTypeGeoKey = 1 (RasterPixelIsArea)
  GeographicTypeGeoKey = 4326 (GCS_WGS_84)
```

B.8.2. Less Common Examples

Unrectified Aerial photo, known tiepoints, in degrees.

We have an aerial photo, and know only the WGS 84 GPS location of several points in the scene: the upper left corner is 120 degrees West, 32 degrees North, the lower-left corner is at 120 degrees West, 30 degrees 20 minutes North, and the lower-right hand corner of the image is at 116 degrees 40 minutes West, 30 degrees 20 minutes North. The photo is not geometrically corrected, however, and the complete projection is therefore not known.

```
ModelTiepointTag=( 0.0,    0.0, 0.0,    -120.0,    32.0, 0.0,
                  0.0, 1000.0, 0.0,    -120.0, 30.33333, 0.0,
                  1000.0, 1000.0, 0.0, -116.6666667, 30.33333, 0.0)
GeoKeyDirectoryTag:
  GTModelTypeGeoKey = 1 (ModelTypeGeographic)
  GTRasterTypeGeoKey = 1 (RasterPixelIsArea)
  GeographicTypeGeoKey = 4326 (GCS_WGS_84)
```

Remark: Since we have not specified the ModelPixelScaleTag, clients reading this GeoTIFF file are not permitted to infer that there is a simple linear relationship between the raster data and the geographic model coordinate space. The only points that are known to be exact are the ones specified in the tiepoint tag.

Rotated Scanned Map

We have a scanned standard British National Grid, covering the 100km grid zone NZ. Consulting documentation for BNG we find that the southwest corner of the NZ zone has an easting,northing of 400000m, 500000m, relative to the BNG standard false origin. This scanned map has a resolution of 100 meter pixels, and was rotated 90 degrees to fit onto the scanner, so that the southwest corner is now the northwest corner. In this case we must use the ModelTransformation tag rather than the tiepoint/scale pair to map the raster data into model space:

```

ModelTransformationTag = ( 0, 100.0, 0, 400000.0,
                          100.0, 0, 0, 500000.0,
                          0, 0, 0, 0,
                          0, 0, 0, 1)
GeoKeyDirectoryTag:
  GTModelTypeGeoKey = 1 ( ModelTypeProjected)
  GTRasterTypeGeoKey = 1 (RasterPixelIsArea)
  ProjectedCSTypeGeoKey = 27700 (PCS_British_National_Grid)
  PCSCitationGeoKey = "British National Grid, Zone NZ"

```

Remark: the matrix has 100.0 in the off-diagonals due to the 90 degree rotation; increasing I points north, and increasing J points east.

Digital Elevation Model

The DMA stores digital elevation models using an equirectangular projection, so that it may be indexed with WGS 84 geographic coordinates. Since elevation postings are point-values, the pixels should not be considered as filling areas, but as point-values at grid vertices. To accommodate the base elevation of the Angeles Crest forest, the pixel value of 0 corresponds to an elevation of 1000 meters relative to WGS 84 reference ellipsoid. The upper left corner is at 120 degrees West, 32 degrees North, and has a pixel scale of 0.2 degrees/pixel longitude, 0.1 degrees/pixel latitude.

```

ModelTiepointTag=(0.0, 0.0, 0.0, -120.0, 32.0, 1000.0)
ModelPixelScale = (0.2, 0.1, 1.0)
GeoKeyDirectoryTag:
  GTModelTypeGeoKey = 2 (ModelTypeGeographic)
  GTRasterTypeGeoKey = 2 (RasterPixelIsPoint)
  GeographicTypeGeoKey = 4326 (GCS_WGS_84)
  VerticalCSTypeGeoKey = 5030 (VertCS_WGS_84_ellipsoid)
  VerticalCitationGeoKey = "WGS 84 Ellipsoid"
  VerticalUnitsGeoKey = 9001 (Linear_Meter)

```

Remarks:

1. Note the "RasterPixelIsPoint" raster space, indicating that the DEM posting of the first pixel is at the raster point (0,0,0), and therefore corresponds to 120W,32N exactly.
2. The third value of the "PixelScale" is 1.0 to indicate that a single pixel-value unit corresponds to 1 meter, and the last tiepoint value indicates that base value zero indicates 1000m above the reference surface.

Annex C: Revision History

Date	Release	Editor	Primary clauses modified	Description
2016-04-28	0.1	G. Editor	all	initial version
2018-06-07	0.2	C. Heazel	all	AsciiDoc version

Annex D: Bibliography

Association Adobe Developers. (1992). *Revision 6.0*.

GeoTIFF profile for Georeferenced Imagery, DGIWG 108, 2014-03-17, Retrieved from https://portal.dgiwg.org/files/?artifact_id=5440&format=pdf

Ritter, N., & Ruth, M. (1997). The GeoTiff data interchange standard for raster geographic images. *International Journal of Remote Sensing*, 18(7), 1637–1647. doi:10.1080/014311697218340Ritter

Niles, &  Ruth, M. (1995). GeoTIFF Format Specification

Wiggins, R. H., Davidson, H. C., Harnsberger, H. R., Lauman, J. R., & Goede, P. a. (2001). Image file formats: past, present, and future. *Radiographics : a review publication of the Radiological Society of North America, Inc*, 21(3), 789–98. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/11353125>