GeoTIFF Format Specification

GeoTIFF Revision 1.0

```
+----------------------------------------------------------------------
--+
```

Specification Version: 1.8.1
Last Modified: 31 October, 1995

Authors:

Niles Ritter, Jet Propulsion Laboratory
Cartographic Applications Group
4800 Oak Grove Dr.
Pasadena, CA 91109
email:ndr@tazboy.jpl.nasa.gov

Mike Ruth, SPOT Image Corp
Product Development Group
1897 Preston White Dr.
Reston, VA 22091
email:ruth@spot.com

Acknowledgments:

GeoTIFF Working Group:
Mike Ruth, Niles Ritter, Ed Grissom, Brett Borup, George Galang,
John Haller, Gary Stephenson, Steve Covington, Tim Nagy,
Jamie Moyers, Jim Stickley,Joe Messina, Yves Somer.

Additional advice from discussions with Tom Lane, Sam Leffler regarding TIFF implementations.

Roger Lott, Fredrik Lundh, and Jarle Land provided valuable information regarding projections, projection code databases and geodetics.

GeoTIFF Mailing list:
Posting: geotiff@tazboy.jpl.nasa.gov
Subscription: geotiff-request@tazboy.jpl.nasa.gov
(send message "subscribe geotiff your-name-here").

Disclaimers and Notes for This Version:

This proposal has not been approved by SPOT, JPL, or any other organization. This represents a proposal, which derives from many discussions between an international body of TIFF users and developers.

The authors and their sponsors assume no liability for any special, incidental, indirect or consequences of any kind, or any damages whatsoever resulting from loss of use, data or profits, whether or not advised of the possibility of damage, and on any theory of liability, arising out of or in connection with the use of this specification.

Copyright

Licenses and Trademarks

Concurrence

The following members of the GeoTIFF working group have reviewed and approved of this revision.

```
   Name                    Organization              Representing
   --------------------    ----------------------    ------------
   Niles Ritter            Jet Propulsion Labs       JPL Carto Group
   Mike Ruth               SPOT Image Corp. (USA)     SPOT Image Corp.
(USA)

   +-------------------------------------------------------------------
   -+
```

Table of Contents

+--
-+

+--
-+

1 Introduction

+--
-+

+-----------------------------------+

1.1 About this Specification

This is a description of a proposal to specify the content and structure of a group of industry-standard tag sets for the management of georeference or geocoded raster imagery using Aldus-Adobe's public domain Tagged-Image File Format (TIFF).

This specification closely follows the organization and structure of the TIFF specification document.

+-----------------------------------+

1.1.1 Background

TIFF has emerged as one of the world's most popular raster file formats. But TIFF remains limited in cartographic applications, since no publicly available, stable structure for conveying geographic information presently exists in the public domain.

Several private solutions exist for recording cartographic information in TIFF tags. Intergraph has a mature and sophisticated geotie tag implementation, but this remains within the private TIFF tagset registered exclusively to Intergraph. Other companies (such as ESRI, and Island Graphics) have geographic solutions which are also proprietary or limited by specific application to their software's architecture.

Many GIS companies, raster data providers, and their clients have requested that the companies concerned with delivery and exploitation of raster geographic imagery develop a publicly available, platform interoperable standard for the support of geographic TIFF imagery. Such TIFF imagery would originate from satellite imaging platforms, aerial platforms, scans of aerial photography or paper maps, or as a result of geographic analysis. TIFF images which were supported by the public "geotie" tagset would be able to be read and positioned correctly in any GIS or digital mapping system which supports the "GeoTIFF" standard, as proposed in this document.

The savings to the users and providers of raster data and exploitation softwares are potentially significant. With a platform interoperable GeoTIFF file, companies could stop spending excessive development resource in support of any and all proprietary formats which are invented. Data providers may be able to produce off-the-shelf imagery products which can be delivered in the "generic" TIFF format quickly and possibly at lower cost. End-users will have the advantage of developed software that exploits the GeoTIFF tags transparently. Most importantly, the same raster TIFF image which can be read and modified in one GIS environment may be equally exploitable in another GIS environment without requiring any file duplication or import/export operation.

+-----------------------------------+

1.1.2 History

The initial efforts to define a TIFF "geotie" specification began under the leadership of Ed Grissom at Intergraph, and others in the early 1990's. In 1994 a formal GeoTIFF mailing-list was created and maintained by Niles Ritter at JPL, which quickly grew to over 140 subscribers from government and industry. The purpose of the list is to discuss common goals and interests in developing an industry-wide GeoTIFF standard, and culminated in a conference in March of 1995 hosted by SPOT Image, with representatives from USGS, Intergraph, ESRI, ERDAS, SoftDesk, MapInfo, NASA/JPL, and others, in which the current working proposal for GeoTIFF was outlined. The outline was condensed into a prerelease GeoTIFF specification document by Niles Ritter, and Mike Ruth of SPOT Image.

Following discussions with Dr. Roger Lott of the European Petroleum Survey Group (EPSG), the GeoTIFF projection parametrization method was extensively modified, and brought into compatibility with both the POSC Epicentre model, and the Federal Geographic Data Committee (FGDC) metadata approaches.

+-----------------------------------+

1.1.3 Scope

The GeoTIFF spec 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.

GeoTIFF does not intend to become a replacement for existing geographic data interchange standards, such as the USGS SDTS standard or the FGDC metadata standard. Rather, it aims to augment an existing popular raster-data format to support georeferencing and geocoding information.

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.

+-----------------------------------+

1.1.4 Features

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 which 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.

+-----------------------------------+

1.2 Revision Notes

This is the final release of GeoTIFF Revision 1.0, supporting the new EPSG 2.x codes.

Changes from 1.8 document: minor spelling and typo corrections.

+-----------------------------------+

1.2.1 Revision Nomenclature

A Revision of GeoTIFF specifications will be denoted by two integers separated by a decimal, indicating the Major and Minor revision numbers. GeoTIFF stores most of its information using a "Key-Code" pairing system; the Major revision number will only be incremented when a substantial addition or modification is made to the list of information Keys, while the Minor Revision number permits incremental augmentation of the list of valid codes.

+-----------------------------------+

1.2.2 New Features

Revision 1.0 New Transformation Matrix Tag.

Index Table added in Section 6.4 to assist in looking up geodesy codes.

+-----------------------------------+

1.2.3 Clarifications

Revision 1.0:

```
 o The former ModelTransformationTag (33920) conflicts with
   an internal Intergraph implementation and is being deprecated,
   in favor of a new tag (34264, registered to JPL).

 o The "Origin" keys have been renamed with "Natural" or "Nat"
   prefixes, to distinguish from "False" origins, and to have
   a closer match to EPSG/POSC terminology. All Revision 0.2
   names shall be recognized in a backward-compatible fashion.

 o The GeoTIFF/Cartlab web page addresses have been moved out
   of the author's ~ndr/ personal directory, and may now be found at:

    http://www-mipl.jpl.nasa.gov/cartlab/geotiff/geotiff.html
```

Revision 0.2:

o South Oriented Gauss Conformal is Transverse Mercator with South pointing up, and so has been given a distinct code, rather than aliased to Transverse Mercator.

Revision 0.1:

o GeoTIFF-writers shall store the GeoKey entries in key-sorted order within the GeoKeyDirectoryTag. This is a change from preliminary discussions which permitted arbitrary order, and more closely follows the TIFF discipline.

o The third value "ScaleZ" in ModelPixelScaleTag = (ScaleX, ScaleY, ScaleZ) shall by default be set to 0, not 1, as suggested in preliminary discussions. This is because most standard model spaces are 2-dimensional (flat), and therefore its vertical shape is independent of the pixel-value.

o The code 32767 shall be used to imply "user-defined", rather than 16384. This avoids breaking up the reserved public GeoKey code space into two discontiguous ranges, 0-16383 and 16385-32767.

o If a GeoKey is coded "undefined", then it is exactly that; no parameters should be provided (e.g. EllipsoidSemiMajorAxis, etc). To provide parameters for a non-coded attribute, use "user-defined".

+----------------------------------+

1.2.4 Organizational changes

None.

+----------------------------------+

1.2.5 Changes in Requirements

Changes to this preliminary revision:

o Support for new transformation matrix tag (34264) required.

+----------------------------------+

1.2.6 Agenda for Future Development

Revision 1.0, which is the first true "Baseline" revision, is proposed to support well-documented, public, relatively simple Projected Coordinate Systems (PCS), including most commonly used and supported in the international public domains today, together with their underlying map-projection systems. Following the critiques of the 0.x Revision phase, the 1.0 Revision spec is hereby released in Sept '95.

In the coming year, incremental 1.x augmentations to the "codes" list will be established, as well as discussions regarding the future "2.0" requirements.

The Revision 2.0 phase is proposed to extend the capability of the GeoTIFF tagsets beyond PCS projections into more complex map projection geometries, including single-project, single-vendor, or proprietary cartographic solutions.

TBD: Sounding Datums and related parameters for Digital Elevation Models (DEM's) and bathymetry -- Revision 2?

+-----------------------------------+

1.3 Administration

+-----------------------------------+

1.3.1 Information and Support:

The most recent version of the GeoTIFF spec, EPSG/POSC tables, and source code is available via anonymous FTP at:

ftp://mtritter.jpl.nasa.gov/pub/tiff/geotiff/

and is mirrored at the USGS:

ftp://ftpmcmc.cr.usgs.gov/release/geotiff/jpl_mirror/

There are several subdirectories called spec/ tables/ and code/.

The USGS also has an archive of prototype GeoTIFF images at:

ftp://ftpmcmc.cr.usgs.gov/release/geotiff/images/

Information and a hypertext version of the GeoTIFF spec is available via WWW at the following site:

http://www-mipl.jpl.nasa.gov/cartlab/geotiff/geotiff.html

A mailing-list is currently active to discuss the on-going development of this standard. To subscribe to this list, send e-mail to:

GeoTIFF-request@tazboy.jpl.nasa.gov

with no subject and the body of the message reading:

```
  subscribe geotiff  your-name-here
```

To post inquiries directly to the list, send email to:

```
    geotiff@tazboy.jpl.nasa.gov
```

+-----------------------------------+

1.3.2 Private Keys and Codes:

As with TIFF, in GeoTIFF private "GeoKeys" and codes may be used, starting with 32768 and above. Unlike the TIFF spec, however, these private key-spaces will not be reserved, and are only to be used for private, internal purposes.

+-----------------------------------+

1.3.3 Proposed Revisions to GeoTIFF

Should a feature arise which is not currently supported, it should be formally proposed for addition to the GeoTIFF spec, through the official mailing-list.

The current maintainer of the GeoTIFF specification is Niles Ritter, though this may change at a later time. Projection codes are maintained through EPSG/POSC, and a mechanism for change/additions will be established through the GeoTIFF mailing list.

+---
-+

2 Baseline GeoTIFF

+---
-+

+-----------------------------------+

2.1 Notation

This spec follows the notation remarks of the TIFF 6.0 spec, regarding "is", "shall", "should", and "may"; the first two indicate mandatory requirements, "should" indicates a strong recommendation, while "may" indicates an option.

+-----------------------------------+

2.2 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:

o 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, implementors of TIFF readers would be more willing to honor a new tag specification if it does not require parsing novel binary structures.

o Any Tag value which is to be used as a "keyword" switch or modifier should be a SHORT type, rather than an ASCII string. This avoids common 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.

o 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.

+-----------------------------------+

2.3 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" in section 5 for the location.

+---------------------------------+

2.4 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 .

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:

```
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 (87BO.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 CoordSystemInfoTag.

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 will be used extensively in section 2.7, where the numerous parameters for defining Coordinate Systems and their underlying projections are defined.

+--------------------------------+

2.5 Coordinate Systems in GeoTIFF

Geotiff has been designed so that standard map coordinate system definitions can be readily stored in a single registered TIFF tag. It has also been designed to allow the description of coordinate system definitions which are non-standard, and for the description of transformations between coordinate 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.

+--------------------------------+

2.5.1 Device Space and GeoTIFF

In standard TIFF 6.0 there are tags which 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.

+----------------------------------+

2.5.2 Raster Coordinate Systems

+----------------------------------+

2.5.2.1 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 of 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.

2.5.2.2 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 are 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).

+-----------------------------------+

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

Projected coordinates, local grid coordinates, and (usually) geographical 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 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

+----------------------------------+

2.5.3.1 Geographic Coordinate Systems

Geographic Coordinate 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 semimajor and semiminor 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-median 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 which 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 Systems

In summary, geographic coordinates are only unique if qualified by the code of the geographic coordinate 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 systems and their component ellipsoids, geodetic datums and prime meridians. Within the Geotiff standard a geographic coordinate system can be identified either by

        the code of a standard geographic coordinate system

or by

a user-defined system.

The user is expected to provide geographic coordinate 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.

+-----------------------------------+

2.5.3.2 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 systems. (Coordinate system transformations are described in section 2.6 below).

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

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

or

through a user-defined name.

+-----------------------------------+

2.5.3.3 Projected Coordinate Systems

Although a geographical coordinate 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 coordinate system transformation method and a set of defining parameters. A projected coordinate 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 system.

In GeoTIFF PCS's are defined using the POSC/EPSG system, in which the PCS planar coordinate system, the Geographic coordinate system, and the transformation between them, are broken down into simpler logical components. Here are schematic formulas showing how the Projected Coordinate Systems and Geographic Coordinates 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/POSC, 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 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 system; the ProjectedCSTypeGeoKey code value alone is sufficient to identify the standard coordinate systems.

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

the code of a standard projected coordinate system

or by

a user-defined system.

User-define projected coordinate systems may be defined by defining the Geographic Coordinate System, the coordinate transformation method and its associated parameters, as well as the planar system's linear units.

2.5.3.4 Vertical Coordinate Systems

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

+----------------------------------+

2.5.4 Reference Parameters

Most of the numerical coding systems and coordinate system definitions are based on the hierarchical system developed by EPSG/POSC. The complete set of EPSG tables used in GeoTIFF is available at:

  ftp://ftpmcmc.cr.usgs.gov/release/geotiff/jpl-mirror/tables

or:

  ftp://mtritter.jpl.nasa.gov/pub/tiff/geotiff/tables

Appended below is the README.TXT file that accompanies the tables of defining parameters for those codes:

```
+------------------------------------+
|      EPSG Geodesy Parameters       |
|    version 2.1, 2nd June 1995.     |
+------------------------------------+
```

The European Petroleum Survey Group (EPSG) has compiled and is distributing this set of parameters defining various geodetic and cartographic coordinate systems to encourage standardisation across the Exploration and Production segment of the oil industry. The data is included as reference data in the Geotiff data exchange specification, in Iris21 the Petroconsultants data model, and in Epicentre, the POSC data model. Parameters map directly to the POSC Epicentre model v2.0, except for data item codes which are included in the files for data management purposes. Geodetic datum parameters are embedded within the geographic coordinate system file. This has been done to ease parameter maintenance as there is a high correlation between geodetic datum names and geographic coordinate system names. The Projected Coordinate System v2.0 tabulation consists of systems associated with locally used projections. Systems utilising the popular UTM grid system have also been included.

Criteria used for material in these lists include:

- information must be in the public domain: "private" data is not included.
- data must be in current use.
- parameters are given to a precision consistent with coordinates being to a precision of one centimetre.

Data is distributed on MS-DOS formatted diskette in comma-separated record format. Additional copies may be obtained from Jean-Patrick Girbig at the address below at a cost of US$100 to cover media and shipping, payment to be made in favour of Petroconsultants S.A at Union Banque Suisses, 1211 Geneve 11, Switzerland (compte number 403 458 60 K).

The data is to be made available on a bulletin board shortly.

Shipping List

```
------------

This data set consists of 8 files:

PROJCS.CSV  Tabulation of Projected Coordinate Systems to
            which map grid coordinates may be referenced.

GEOGCS.CSV  Tabulation of Geographic Coordinate Systems to
            which latitude and longitude coordinates may be
            referenced.  This table includes the equivalent
            geocentric coordinate systems and also the
            geodetic datum, reference to which allows latitude
            and longitude or geocentric XYZ to uniquely
            describe a location on the earth.

VERTCS.CSV  Tabulation of Vertical Coordinate Systems to
            which heights or depths may be referenced. This
            table is currently in an early form.

PROJ.CSV    Tabulation of transformation methods and
            parameters through which Projected Coordinate
            Systems are defined and related to Geographic
            Coordinate Systems.

ELLIPS.CSV  Tabulation of reference ellipsoids upon which
            geodetic datums are based.

PMERID.CSV  Tabulation of prime meridians upon which geodetic
            datums are based.

UNITS.CSV   Tabulation of length units used in Projected and
            Vertical Coordinate Systems and angle units used
            in Geographic Coordinate Systems.

README.TXT  This file.
```

```
+-------------------------------------------------------------------
--+
```

2.6 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.

+----------------------------------+

2.6.1 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:

```
|-   -|     |-                  -|  |-   -|
|  X  |     |   a   b   0   d    |  |  I  |
|     |     |                    |  |     |
|  Y  |     |   e   f   0   h    |  |  J  |
|     |  =  |                    |  |     |
|  Z  |     |   0   0   0   0    |  |  K  |
|     |     |                    |  |     |
|  1  |     |   0   0   0   1    |  |  1  |
|-   -|     |-                  -|  |-   -|
```

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:

```
|-                           -|
|   Sx     0.0   0.0   Tx     |
|                             |      Tx = X - I/Sx
|   0.0  -Sy     0.0   Ty     |      Ty = Y + J/Sy
|                             |      Tz = Z - K/Sz  (if not 0)
|   0.0   0.0   Sz     Tz     |
|                             |
|   0.0   0.0   0.0   1.0     |
|-                           -|
```

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 be contain valid GeoTIFF matrix information if the tag-count is 16; the Intergraph version uses 17 values.

+-----------------------------------+

2.6.2 Coordinate Transformation Data Flow

The dataflow of the various GeoTIFF parameter datasets is based upon the EPSG/POSC configuration. Here is the text of the description accompanying the EPSG parameter tables:

The data files (.CSV) have a hierarchical structure:

```
 +---------------------------+   +----------------------------+
 |           VERTCS          |   |           PROJCS           |
 +---------------------------+   +----------------------------+
 |Vertical Coordinate Systems|   |Projected Coordinate Systems|
 +-------------+-------------+   +------------+---------------+
               |                              |
      +--------+                              |
      |                                       |
      |            +--------------------------+
      |            |                          |
      |            |            +-------------+---------------+
      |            |            |            GEOGCS           |
      |            |            +-----------------------------+
      |            |            |Geographic Coordinate Systems|
      |            |            |Geocentric Coordinate Systems|
      |            |            +-----------------------------+
      |            |            |       Geodetic Datums       |
      |            |            +-------------+---------------+
      |            |                          |
      |            |                 +--------+-------+
      |            |                 |                |
      |     +------+-----+    +------+-----+   +------+-------+
      |     |    PROJ    |    |   ELLIPS   |   |    PMERID    |
      |     +------------+    +------------+   +--------------+
      |     | Projection |    | Ellipsoid  |   |Prime Meridian|
      |     | Parameters |    | Parameters |   |  Parameters  |
      |     +------+-----+    +------+-----+   +------+-------+
      |            |                 |                |
      +------------+-----------+-----+----------------+
                               |
                 +-------------+------------+
                 |           UNITS          |
                 +--------------------------+
                 | Linear and Angular Units |
                 +--------------------------+
```

The parameter listings are "living documents" and will be updated by the EPSG from time to time. Any comment or suggestions for improvements should be directed to:

Jean-Patrick Girbig, Manager Cartography, Petroconsultants S.A., PO Box 152, 24 Chemin de la Marie, 1258 Perly-Geneva, Switzerland.	or	Roger Lott, Head of Survey, BP Exploration, Uxbridge One, Harefield Road, Uxbridge, Middlesex UB8 1PD, England. Internet: lottrj@txpcap.hou.xwh.bp.com

Requests for the inclusion of new data should include supporting documentation. Requests for changing existing data should include reference to both the name and code of the item.

+---------------------------------+

2.6.3 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 system (PCS). If you are geocoding this data set, then the model space is defined to be the corresponding geographic, geocentric or Projected coordinate system (skip to the "Cookbook" section 2.7.3 first to do determine this).

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

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

+-----------------------------------+

2.7 Geocoding Raster Data

+-----------------------------------+

2.7.1 General Approach

A geocoded image is a georeferenced image as described in section 2.6, which also specifies a model space coordinate 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:

```
     ModelPixelScaleTag
     ModelTiepointTag                GeoKeyDirectoryTag CS
 R  -------- OR ---------------> M  --------- AND  -----------> Earth
     ModelTransformationTag          GeoDoubleParamsTag
                                     GeoAsciiParamsTag
```

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.

+----------------------------------+

2.7.2 GeoTIFF GeoKeys for Geocoding

As mentioned above, all information regarding the Model Coordinate System used in the raster data is referenced from the GeoKeyDirectoryTag, which stores all of the GeoKey entries. In the Appendix, section 6.2 summarizes all of the GeoKeys defined for baseline GeoTIFF, and their corresponding codes are documented in section 6.3. Only the Keys themselves are documented here.

+----------------------------------+

Common Features

+----------------------------------+

Public and Private Key and Code Ranges

GeoTIFF GeoKey ID's may take any value between 0 and 65535. Following TIFF general approach, the GeoKey ID's from 32768 and above are available for private implementations. However, no registry will be established for these keys or codes, so developers are warned to use them at their own risk.

The Key ID's from 0 to 32767 are reserved for use by the official GeoTIFF spec, and are broken down into the following sub-domains:

```
[    0,  1023]       Reserved
[ 1024,  2047]       GeoTIFF Configuration Keys
[ 2048,  3071]       Geographic/Geocentric CS Parameter Keys
[ 3072,  4095]       Projected CS Parameter Keys
[ 4096,  5119]       Vertical CS Parameter Keys
[ 5120, 32767]       Reserved
[32768, 65535]       Private use
```

GeoKey codes, like keys and tags, also range from 0 to 65535. Following the TIFF approach, all codes from 32768 and above are available for private user implementation. There will be no registry for these codes, however, and so developers must be sure that these tags will only be used internally. Use private codes at your own risk.

The codes from 0 to 32767 for all public GeoKeys are reserved by this GeoTIFF specification.

Common Public Code Values

For consistency, several key codes have the same meaning in all implemented GeoKeys possessing a SHORT numerical coding system:

```
       0 = undefined
   32767 = user-defined
```

The "undefined" code means that this parameter is intentionally omitted, for whatever reason. For example, the datum used for a given map may be unknown, or the accuracy of a aerial photo is so low that to specify a particular datum would imply a higher accuracy than is in the data.

The "user-defined" code means that a feature is not among the standard list, and is being explicitly defined. In cases where this is meaningful, Geokey parameters have been supplied for the user to define this feature.

"User-Defined" requirements: In each section below a specification of the additional GeoKeys required for the "user-defined" option is given. In all cases the corresponding "Citation" key is strongly recommended, as per the FGDC Metadata standard regarding "local" types.

+-----------------------------------+

GeoTIFF Configuration GeoKeys

+-----------------------------------+

These keys are to be used to establish the general configuration of this file's coordinate system, including the types of raster coordinate systems, model coordinate systems, and citations if any.

+---
--+

GTModelTypeGeoKey

Key ID = 1024
Type: SHORT (code)
Values: Section 6.3.1.1 Codes

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.

GeoKey requirements for User-Defined Model Type (not advisable):

GTCitationGeoKey

+---+

GTRasterTypeGeoKey

Key ID = 1025
Type = Section 6.3.1.2 codes

This establishes the Raster Space coordinate system used; 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.

+---+

GTCitationGeoKey

Key ID = 1026
Type = ASCII

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.

+---+

+---------------------------------+

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.

+---+

GeographicTypeGeoKey

Key ID = 2048

Type = SHORT (code)
Values = Section 6.3.2.1 Codes

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.

GeoKey Requirements for User-Defined geographic CS:

GeogCitationGeoKey
GeogGeodeticDatumGeoKey
GeogAngularUnitsGeoKey (if not degrees)
GeogPrimeMeridianGeoKey (if not Greenwich)

+---+

GeogCitationGeoKey

Key ID = 2049
Type = ASCII
Values = text

General citation and reference for all Geographic CS parameters.

+---+

GeogGeodeticDatumGeoKey

Key ID = 2050
Type = SHORT (code)
Values = Section 6.3.2.2 Codes

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.

GeoKey Requirements for User-Defined Horizontal Datum:
GeogCitationGeoKey
GeogEllipsoidGeoKey

+---+

GeogPrimeMeridianGeoKey

Key ID = 2051
Type = SHORT (code)
Units: Section 6.3.2.4 code

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

+---+

GeogPrimeMeridianLongGeoKey

Key ID = 2061
Type = DOUBLE
Units = GeogAngularUnits

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

+--+

GeogLinearUnitsGeoKey

Key ID = 2052
Type = DOUBLE
Values: Section 6.3.1.3 Codes

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

+--+

GeogLinearUnitSizeGeoKey

Key ID = 2053
Type = DOUBLE
Units: meters

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

+--+

GeogAngularUnitsGeoKey

Key ID = 2054
Type = SHORT (code)
Values = Section 6.3.1.4 Codes

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

GeoKey Requirements for "user-defined" units:
GeogCitationGeoKey
GeogAngularUnitSizeGeoKey

+--+

GeogAngularUnitSizeGeoKey

Key ID = 2055
Type = DOUBLE
Units: radians

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

+--+

GeogEllipsoidGeoKey

Key ID = 2056
Type = SHORT (code)
Values = Section 6.3.2.3 Codes

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

GeoKey Requirements for User-Defined Ellipsoid:

```
GeogCitationGeoKey
[GeogSemiMajorAxisGeoKey,
        [GeogSemiMinorAxisGeoKey | GeogInvFlatteningGeoKey] ]
```

+---+

GeogSemiMajorAxisGeoKey

Key ID = 2057
Type = DOUBLE
Units: Geocentric CS Linear Units

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

+---+

GeogSemiMinorAxisGeoKey

Key ID = 2058
Type = DOUBLE
Units: Geocentric CS Linear Units

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

+---+

GeogInvFlatteningGeoKey

Key ID = 2059
Type = DOUBLE
Units: none.

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.

+---+

GeogAzimuthUnitsGeoKey

Key ID = 2060
Type = SHORT (code)
Values = Section 6.3.1.4 Codes

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.

+---+

+-----------------------------------+

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.

+---+

ProjectedCSTypeGeoKey

Key ID = 3072
Type = SHORT (codes)
Values: Section 6.3.3.1 codes

This code is provided to specify the projected coordinate system.

GeoKey requirements for "user-defined" PCS families:
PCSCitationGeoKey
ProjectionGeoKey

+---+

PCSCitationGeoKey

Key ID = 3073
Type = ASCII

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.

+---+

+-----------------------------------+

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.

+---+

ProjectionGeoKey

Key ID = 3074
Type = SHORT (code)
Values: Section 6.3.3.2 codes

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.

GeoKeys Required for "user-defined" Projections:

PCSCitationGeoKey
ProjCoordTransGeoKey
ProjLinearUnitsGeoKey
(additional parameters depending on ProjCoordTransGeoKey).

+---+

ProjCoordTransGeoKey

Key ID = 3075
Type = SHORT (code)
Values: Section 6.3.3.3 codes

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.

GeoKeys Required for "user-defined" Coordinate Transformations:

PCSCitationGeoKey
<additional parameter geokeys depending on the Coord. Trans. specified).

+---+

ProjLinearUnitsGeoKey

Key ID = 3076
Type = SHORT (code)
Values: Section 6.3.1.3 codes

Defines linear units used by this projection.

+--+

ProjLinearUnitSizeGeoKey

```
Key ID = 3077
Type = DOUBLE
Units: meters
```

Defines size of user-defined linear units in meters.

+--+

ProjStdParallel1GeoKey

```
Key ID = 3078
Type = DOUBLE
Units: GeogAngularUnit
Alias: ProjStdParallelGeoKey (from Rev 0.2)
```

Latitude of primary Standard Parallel.

+--+

ProjStdParallel2GeoKey

```
Key ID = 3079
Type = DOUBLE
Units: GeogAngularUnit
```

Latitude of second Standard Parallel.

+--+

ProjNatOriginLongGeoKey

```
Key ID = 3080
Type = DOUBLE
Units: GeogAngularUnit
Alias: ProjOriginLongGeoKey
```

Longitude of map-projection Natural origin.

+--+

ProjNatOriginLatGeoKey

```
Key ID = 3081
Type = DOUBLE
Units: GeogAngularUnit
Alias: ProjOriginLatGeoKey
```

Latitude of map-projection Natural origin.

+--+

ProjFalseEastingGeoKey

```
Key ID = 3082
Type = DOUBLE
Units: ProjLinearUnit
```

Gives the easting coordinate of the map projection Natural origin.

+---+

ProjFalseNorthingGeoKey

Key ID = 3083
Type = DOUBLE
Units: ProjLinearUnit

Gives the northing coordinate of the map projection Natural origin.

+---+

ProjFalseOriginLongGeoKey

Key ID = 3084
Type = DOUBLE
Units: GeogAngularUnit

Gives the longitude of the False origin.

+---+

ProjFalseOriginLatGeoKey

Key ID = 3085
Type = DOUBLE
Units: GeogAngularUnit

Gives the latitude of the False origin.

+---+

ProjFalseOriginEastingGeoKey

Key ID = 3086
Type = DOUBLE
Units: ProjLinearUnit

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

+---+

ProjFalseOriginNorthingGeoKey

Key ID = 3087
Type = DOUBLE
Units: ProjLinearUnit

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

+---+

ProjCenterLongGeoKey

Key ID = 3088
Type = DOUBLE
Units: GeogAngularUnit

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

+---+

ProjCenterLatGeoKey

Key ID = 3089
Type = DOUBLE
Units: GeogAngularUnit

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

+---+

ProjCenterEastingGeoKey

Key ID = 3090
Type = DOUBLE
Units: ProjLinearUnit

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

+---+

ProjFalseOriginNorthingGeoKey

Key ID = 3091
Type = DOUBLE
Units: ProjLinearUnit

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

+---+

ProjScaleAtNatOriginGeoKey

Key ID = 3092
Type = DOUBLE
Units: none
Alias: ProjScaleAtOriginGeoKey (Rev. 0.2)

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

+---+

ProjScaleAtCenterGeoKey

Key ID = 3093
Type = DOUBLE
Units: none

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

+---+

ProjAzimuthAngleGeoKey

Key ID = 3094
Type = DOUBLE
Units: GeogAzimuthUnit

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.

+---+

ProjStraightVertPoleLongGeoKey

Key ID = 3095
Type = DOUBLE
Units: GeogAngularUnit

Longitude at Straight Vertical Pole. For polar stereographic.

+---+

GeogAzimuthUnitsGeoKey

Key ID = 2060
Type = SHORT (code)
Values = Section 6.3.1.4 Codes

This key is actually part of the "Geographic CS Parameter Keys" section, but is mentioned here as it is useful for defining units used in the azimuthal projection parameters.

+---+

+-----------------------------------+

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.

+---+

VerticalCSTypeGeoKey

Key ID = 4096
Type = SHORT (code)
Values = Section 6.3.4.1 Codes

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

+--+

VerticalCitationGeoKey

Key ID = 4097
Type = ASCII
Values = text

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

+--+

VerticalDatumGeoKey

Key ID = 4098
Type = SHORT (code)
Values = Section 6.3.4.2 codes

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

+--+

VerticalUnitsGeoKey

Key ID = 4099
Type = SHORT (code)
Values = Section 6.3.1.3 Codes

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.

+----------------------------------+

2.7.3 Cookbook for Geocoding Data

Step 1: Determine the Coordinate system type of the raster data, based on
the nature of the data: pixels derived from scanners or other optical devices represent areas, and most commonly will use the RasterPixelIsArea coordinate system. Pixel data such as digital elevation models represent points, and will probably use RasterPixelIsPoint coordinates.

Store in: GTRasterTypeGeoKey

Step 2: Determine which class of model space coordinates are most natural
for this dataset:Geographic, Geocentric, or Projected Coordinate
System. Usually this will be PCS.

Store in: GTModelTypeGeoKey

Step 3: This step depends on the GTModelType:

case PCS: Determine the PCS projection system. Most of the PCS's used in standard State Plane and national grid systems are defined, so check this list first; the EPSG index in section 6.4 may be useful for this purpose.

Store in: ProjectedCSTypeGeoKey, ProjectedCSTypeGeoKey

If coded, it will not be necessary to specify the Projection datum, etc for this case, since all of those parameters are determined by the ProjectedCSTypeGeoKey code. Skip to step 4 from here.

If none of the coded PCS's match your system, then this is a user-defined PCS. Use the Projection code list to check for standard projection systems.

Store in: ProjectionGeoKey and skip to Geographic CS case.

If none of the Projection codes match your system, then this is a user-defined projection. Use the ProjCoordTransGeoKey to specify the coordinate transformation method (e.g. Transverse Mercator), and all of the associated parameters of that method.
Also define the linear units used in the planar coordinate system.

Store in: ProjCoordTransGeoKey, ProjLinearUnitsGeoKey
<and other CT related parameter keys>

Now continue on to define the Geographic CS, below.

case GEOCENTRIC:
case GEOGRAPHIC: Check the list of standard GCS's and use the corresponding code. To use a code both the Datum, Prime Meridian, and angular units must match those of the code.

Store in: GeographicTypeGeoKey and skip to Step 4.

If none of the coded GCS's match exactly, then this is a user-defined GCS. Check the list of standard datums, Prime Meridians, and angular units to define your system.

Store in: GeogGeodeticDatumGeoKey, GeogAngularUnitsGeoKey, GeogPrimeMeridianGeoKey and skip to Step 4.

If none of the datums match your system, you have a user-defined datum, which is an odd system, indeed. Use

```
        the GeogEllipsoidGeoKey to select the appropriate ellipsoid
        or use the GeogSemiMajorAxisGeoKey, GeogInvFlatteningGeoKey
to
        define, and give a reference using the GeogCitationGeoKey.

        Store in: GeogEllipsoidGeoKey, etc. and go to Step 4.


Step 4: Install the GeoKeys/codes into the GeoKeyDirectoryTag, and the
      DOUBLE and ASCII key values into the corresponding value-tags.

Step 5: Having completely defined the Raster & Model coordinate system,
      go to Cookbook section 2.6.2 and use the Georeferencing Tags
      to tie the raster image down onto the Model space.
```

+-----------------------------------+

3 Examples

+-----------------------------------+

Here are some examples of how GeoTIFF may be implemented at the Tag and GeoKey level, following the general "Cookbook" approach above.

+-----------------------------------+

3.1 Common Examples

+-----------------------------------+

3.1.1. UTM Projected Aerial Photo

We have an aerial photo which has been orthorectified and resampled to a UTM grid, zone 60, using WGS84 datum; the coordinates of the upper-left corner of the image is are 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.

+----------------------------------+

3.1.2. 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 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).

+----------------------------------+

3.1.3. 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.

+--
-+

3.1.4. 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)
```

+---------------------------------+

3.2 Less Common Examples

+---------------------------------+

3.2.1. Unrectified Aerial photo, known tiepoints, in degrees.

We have an aerial photo, and know only the WGS84 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 know to be exact are the ones specified in the tiepoint tag.

+---------------------------------+

3.2.2. 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.

+-----------------------------------+

3.2.3. Digital Elevation Model

The DMA stores digital elevation models using an equirectangular projection, so that it may be indexed with WGS84 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 WGS84 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.

+-----------------------------------+

4 Extended GeoTIFF

+--
-+

This section is for future development TBD.

Possible additional GeoKeys for Revision 2.0:

PerspectHeightGeoKey	(General Vertical Nearsided Perspective)	
SOMInclinAngleGeoKey	(SOM)	
SOMAscendLongGeoKey	(SOM)	
SOMRevPeriodGeoKey	(SOM)	
SOMEndOfPathGeoKey	(SOM)	? is this needed ? SHORT
SOMRatioGeoKey	(SOM)	
SOMPathNumGeoKey	(SOM)	SHORT
SOMSatelliteNumGeoKey	(SOM)	SHORT
OEAShapeMGeoKey	(Oblated Equal Area)	
OEAShapeNGeoKey	(Oblated Equal Area)	
OEARotationAngleGeoKey	(Oblated Equal Area)	

Other items for consideration:

o Digital Elevation Model information, such as Vertical Datums, Sounding Datums.

o Accuracy Keys for linear, circular, and spherical errors, etc.

o Source information, such as details of an original coordinate system and of transformations between it and the coordinate system in which data is being exchanged.

+---
-+

5 References

+---
-+

1. EPSG/POSC Projection Coding System Tables. Available via FTP to:

   ftp://mtritter.jpl.nasa.gov/pub/tiff/geotiff/tables

   or its USGS mirror site:

   ftp://ftpmcmc.cr.usgs.gov/release/geotiff/jpl-mirror/tables

2. TIFF Revision 6.0 Specification: A PDF formatted version is available via FTP to:

ftp://ftp.adobe.com/pub/adobe/DeveloperSupport/TechNotes/PDFfiles/TIFF6.pdf

   PostScript formatted text versions available at:.

   ftp://sgi.com/graphics/tiff/TIFF6.ps.Z (compressed)
   ftp://sgi.com/graphics/tiff/TIFF6.ps (uncompressed)

3. LIBGEOTIFF -- Public Domain GeoTIFF library, available via anonymous FTP to:

ftp://mtritter.jpl.nasa.gov/pub/tiff/geotiff/code

or its USGS mirror site:

ftp://ftpmcmc.cr.usgs.gov/release/geotiff/jpl-mirror/code

4. LIBTIFF -- Public Domain TIFF library, available via anonymous FTP to:

   ftp://sgi.com/graphics/tiff/

5. Spatial Data Transfer Standard (SDTS) of the USGS. (Federal Information Processing Standard (FIPS) 173):

   ftp://sdts.er.usgs.gov/pub/sdts/

   SDTS Task Force
   U.S. Geological Survey
   526 National Center
   Reston, VA 22092

   E-mail: sdts@usgs.gov

6. Map use: reading, analysis, interpretation. Muehrcke, Phillip C. 1986. Madison, WI: JP Publications.

7. Map projections: a working manual. Snyder, John P. 1987. USGS Professional Paper 1395. Washington, DC: United States Government Printing Office.

8. Notes for GIS and The Geographer's Craft at U. Texas, on the World Wide Web (WWW) (current as of 10 April 1995):

   http://wwwhost.cc.utexas.edu/ftp/pub/grg/gcraft/notes/notes.html

9. Digital Geographic Information Exchange Standard (DIGEST). Allied Geographic Publication No 3, Edition 1.2 (AGeoP-3) (NATO Unclassified).

10. POSC Petrotechnical Open Software Corporation Web site:

    http://www.posc.org/

+---
-+

6 Appendices

+---
-+

+-----------------------------------+

6.1 Tag ID Summary

Here are all of the TIFF tags (and their owners) that are used to store GeoTIFF information of any type. It is very unlikely that any other tags will be necessary in the future (since most additional information will be encoded as a GeoKey).

```
 ModelPixelScaleTag     = 33550 (SoftDesk)
 ModelTransformationTag = 34264 (JPL Carto Group)
 ModelTiepointTag       = 33922 (Intergraph)
 GeoKeyDirectoryTag     = 34735 (SPOT)
 GeoDoubleParamsTag     = 34736 (SPOT)
 GeoAsciiParamsTag      = 34737 (SPOT)
```

Obsoleted Implementation:

```
  IntergraphMatrixTag = 33920 (Intergraph) -- Use
ModelTransformationTag.
```

+-----------------------------------+

6.2 Key ID Summary

+-----------------------------------+

+-----------------------------------+

6.2.1 GeoTIFF Configuration Keys

```
GTModelTypeGeoKey            = 1024 /* Section 6.3.1.1 Codes       */
GTRasterTypeGeoKey           = 1025 /* Section 6.3.1.2 Codes       */
GTCitationGeoKey             = 1026 /* documentation */
```

+-----------------------------------+

6.2.2 Geographic CS Parameter Keys

```
GeographicTypeGeoKey         = 2048 /* Section 6.3.2.1 Codes     */
GeogCitationGeoKey           = 2049 /* documentation             */
GeogGeodeticDatumGeoKey      = 2050 /* Section 6.3.2.2 Codes     */
GeogPrimeMeridianGeoKey      = 2051 /* Section 6.3.2.4 codes     */
GeogLinearUnitsGeoKey        = 2052 /* Section 6.3.1.3 Codes     */
GeogLinearUnitSizeGeoKey     = 2053 /* meters                    */
GeogAngularUnitsGeoKey       = 2054 /* Section 6.3.1.4 Codes     */
GeogAngularUnitSizeGeoKey    = 2055 /* radians                   */
```

```
GeogEllipsoidGeoKey          = 2056 /* Section 6.3.2.3 Codes     */
GeogSemiMajorAxisGeoKey      = 2057 /* GeogLinearUnits           */
GeogSemiMinorAxisGeoKey      = 2058 /* GeogLinearUnits           */
GeogInvFlatteningGeoKey      = 2059 /* ratio                     */
GeogAzimuthUnitsGeoKey       = 2060 /* Section 6.3.1.4 Codes     */
GeogPrimeMeridianLongGeoKey  = 2061 /* GeogAngularUnit           */
```

+-----------------------------------+

6.2.3 Projected CS Parameter Keys

```
ProjectedCSTypeGeoKey          = 3072  /* Section 6.3.3.1 codes   */
PCSCitationGeoKey              = 3073  /* documentation           */
ProjectionGeoKey               = 3074  /* Section 6.3.3.2 codes   */
ProjCoordTransGeoKey           = 3075  /* Section 6.3.3.3 codes   */
ProjLinearUnitsGeoKey          = 3076  /* Section 6.3.1.3 codes   */
ProjLinearUnitSizeGeoKey       = 3077  /* meters                  */
ProjStdParallel1GeoKey         = 3078  /* GeogAngularUnit */
ProjStdParallel2GeoKey         = 3079  /* GeogAngularUnit */
ProjNatOriginLongGeoKey        = 3080  /* GeogAngularUnit */
ProjNatOriginLatGeoKey         = 3081  /* GeogAngularUnit */
ProjFalseEastingGeoKey         = 3082  /* ProjLinearUnits */
ProjFalseNorthingGeoKey        = 3083  /* ProjLinearUnits */
ProjFalseOriginLongGeoKey      = 3084  /* GeogAngularUnit */
ProjFalseOriginLatGeoKey       = 3085  /* GeogAngularUnit */
ProjFalseOriginEastingGeoKey   = 3086  /* ProjLinearUnits */
ProjFalseOriginNorthingGeoKey  = 3087  /* ProjLinearUnits */
ProjCenterLongGeoKey           = 3088  /* GeogAngularUnit */
ProjCenterLatGeoKey            = 3089  /* GeogAngularUnit */
ProjCenterEastingGeoKey        = 3090  /* ProjLinearUnits */
ProjCenterNorthingGeoKey       = 3091  /* ProjLinearUnits */
ProjScaleAtNatOriginGeoKey     = 3092  /* ratio   */
ProjScaleAtCenterGeoKey        = 3093  /* ratio   */
ProjAzimuthAngleGeoKey         = 3094  /* GeogAzimuthUnit */
ProjStraightVertPoleLongGeoKey = 3095  /* GeogAngularUnit */
```

Aliases:

```
ProjStdParallelGeoKey     = ProjStdParallel1GeoKey
ProjOriginLongGeoKey      = ProjNatOriginLongGeoKey
ProjOriginLatGeoKey       = ProjNatOriginLatGeoKey
ProjScaleAtOriginGeoKey   = ProjScaleAtNatOriginGeoKey
```

+-----------------------------------+

6.2.4 Vertical CS Keys

```
VerticalCSTypeGeoKey           = 4096   /* Section 6.3.4.1 codes   */
VerticalCitationGeoKey         = 4097   /* documentation */
```

```
   VerticalDatumGeoKey            = 4098   /* Section 6.3.4.2 codes   */
   VerticalUnitsGeoKey            = 4099   /* Section 6.3.1.3 codes   */
```

+--+

+-----------------------------------+

6.3 Key Code Summary

+-----------------------------------+

6.3.1 GeoTIFF General Codes

This section includes the general "Configuration" key codes, as well as general codes which are used by more than one key (e.g. units codes).

+-----------------------------------+

6.3.1.1 Model Type Codes

```
Ranges:

   0               = undefined
   [   1,  32766] = GeoTIFF Reserved Codes
   32767          = user-defined
   [32768, 65535] = Private User Implementations

GeoTIFF defined CS Model Type Codes:

   ModelTypeProjected   = 1   /* Projection Coordinate System         */
   ModelTypeGeographic  = 2   /* Geographic latitude-longitude System */
   ModelTypeGeocentric  = 3   /* Geocentric (X,Y,Z) Coordinate System */

Notes:

   1. ModelTypeGeographic and ModelTypeProjected
      correspond to the FGDC metadata Geographic and
      Planar-Projected coordinate system types.
```

+-----------------------------------+

6.3.1.2 Raster Type Codes

```
Ranges:

   0              = undefined
   [   1,  1023] = Raster Type Codes (GeoTIFF Defined)
   [1024, 32766] = Reserved
   32767         = user-defined
   [32768, 65535]= Private User Implementations


Values:
   RasterPixelIsArea  = 1
```

```
   RasterPixelIsPoint = 2
```

Note: Use of "user-defined" or "undefined" raster codes is not recommended.

+--------------------------------+

6.3.1.3 Linear Units Codes

There are several different kinds of units that may be used in geographically related raster data: linear units, angular units, units of time (e.g. for radar-return), CCD-voltages, etc. For this reason there will be a single, unique range for each kind of unit, broken down into the following currently defined ranges:

Ranges:

```
   0             = undefined
   [   1,  2000] = Obsolete GeoTIFF codes
   [2001,  8999] = Reserved by GeoTIFF
   [9000,  9099] = EPSG Linear Units.
   [9100,  9199] = EPSG Angular Units.
   32767         = user-defined unit
   [32768, 65535]= Private User Implementations
```

Linear Unit Values (See the ESPG/POSC tables for definition):

```
   Linear_Meter =     9001
   Linear_Foot =      9002
   Linear_Foot_US_Survey =   9003
   Linear_Foot_Modified_American =  9004
   Linear_Foot_Clarke =      9005
   Linear_Foot_Indian =      9006
   Linear_Link =      9007
   Linear_Link_Benoit =      9008
   Linear_Link_Sears =       9009
   Linear_Chain_Benoit =     9010
   Linear_Chain_Sears =      9011
   Linear_Yard_Sears =       9012
   Linear_Yard_Indian =      9013
   Linear_Fathom =    9014
   Linear_Mile_International_Nautical =    9015
```

+--------------------------------+

6.3.1.4 Angular Units Codes

These codes shall be used for any key that requires specification of an angular unit of measurement.

Angular Units

```
Angular_Radian =   9101
Angular_Degree =   9102
Angular_Arc_Minute =      9103
Angular_Arc_Second =      9104
Angular_Grad =     9105
Angular_Gon =      9106
Angular_DMS =      9107
Angular_DMS_Hemisphere =  9108
```

+----------------------------------+

6.3.2 Geographic CS Codes

+----------------------------------+

6.3.2.1 Geographic CS Type Codes

```
Note: A Geographic coordinate system consists of both a datum and a
Prime Meridian. Some of the names are very similar, and differ only
in the Prime Meridian, so be sure to use the correct one. The codes
beginning with GCSE_xxx are unspecified GCS which use ellipsoid
(xxx); it is recommended that only the codes beginning with GCS_ be
used if possible.
```

Ranges:

```
0 = undefined
[    1,  1000] = Obsolete EPSG/POSC Geographic Codes
[ 1001,  3999] = Reserved by GeoTIFF
[ 4000, 4199]  = EPSG GCS Based on Ellipsoid only
[ 4200, 4999]  = EPSG GCS Based on EPSG Datum
[ 5000, 32766] = Reserved by GeoTIFF
32767          = user-defined GCS
[32768, 65535] = Private User Implementations
```

Values:

```
Note: Geodetic datum using Greenwich PM have codes equal to
the corresponding Datum code - 2000.

GCS_Adindan =      4201
GCS_AGD66 = 4202
GCS_AGD84 = 4203
GCS_Ain_el_Abd =   4204
GCS_Afgooye =      4205
GCS_Agadez =       4206
GCS_Lisbon =       4207
GCS_Aratu = 4208
GCS_Arc_1950 =     4209
GCS_Arc_1960 =     4210
GCS_Batavia =      4211
```

GCS_Barbados = 4212
GCS_Beduaram = 4213
GCS_Beijing_1954 = 4214
GCS_Belge_1950 = 4215
GCS_Bermuda_1957 = 4216
GCS_Bern_1898 = 4217
GCS_Bogota = 4218
GCS_Bukit_Rimpah = 4219
GCS_Camacupa = 4220
GCS_Campo_Inchauspe = 4221
GCS_Cape = 4222
GCS_Carthage = 4223
GCS_Chua = 4224
GCS_Corrego_Alegre = 4225
GCS_Cote_d_Ivoire = 4226
GCS_Deir_ez_Zor = 4227
GCS_Douala = 4228
GCS_Egypt_1907 = 4229
GCS_ED50 = 4230
GCS_ED87 = 4231
GCS_Fahud = 4232
GCS_Gandajika_1970 = 4233
GCS_Garoua = 4234
GCS_Guyane_Francaise = 4235
GCS_Hu_Tzu_Shan = 4236
GCS_HD72 = 4237
GCS_ID74 = 4238
GCS_Indian_1954 = 4239
GCS_Indian_1975 = 4240
GCS_Jamaica_1875 = 4241
GCS_JAD69 = 4242
GCS_Kalianpur = 4243
GCS_Kandawala = 4244
GCS_Kertau = 4245
GCS_KOC = 4246
GCS_La_Canoa = 4247
GCS_PSAD56 = 4248
GCS_Lake = 4249
GCS_Leigon = 4250
GCS_Liberia_1964 = 4251
GCS_Lome = 4252
GCS_Luzon_1911 = 4253
GCS_Hito_XVIII_1963 = 4254
GCS_Herat_North = 4255
GCS_Mahe_1971 = 4256
GCS_Makassar = 4257
GCS_EUREF89 = 4258
GCS_Malongo_1987 = 4259
GCS_Manoca = 4260
GCS_Merchich = 4261
GCS_Massawa = 4262
GCS_Minna = 4263
GCS_Mhast = 4264
GCS_Monte_Mario = 4265
GCS_M_poraloko = 4266
GCS_NAD27 = 4267
GCS_NAD_Michigan = 4268

GCS_NAD83 = 4269
GCS_Nahrwan_1967 = 4270
GCS_Naparima_1972 = 4271
GCS_GD49 = 4272
GCS_NGO_1948 = 4273
GCS_Datum_73 = 4274
GCS_NTF = 4275
GCS_NSWC_9Z_2 = 4276
GCS_OSGB_1936 = 4277
GCS_OSGB70 = 4278
GCS_OS_SN80 = 4279
GCS_Padang = 4280
GCS_Palestine_1923 = 4281
GCS_Pointe_Noire = 4282
GCS_GDA94 = 4283
GCS_Pulkovo_1942 = 4284
GCS_Qatar = 4285
GCS_Qatar_1948 = 4286
GCS_Qornoq = 4287
GCS_Loma_Quintana = 4288
GCS_Amersfoort = 4289
GCS_RT38 = 4290
GCS_SAD69 = 4291
GCS_Sapper_Hill_1943 = 4292
GCS_Schwarzeck = 4293
GCS_Segora = 4294
GCS_Serindung = 4295
GCS_Sudan = 4296
GCS_Tananarive = 4297
GCS_Timbalai_1948 = 4298
GCS_TM65 = 4299
GCS_TM75 = 4300
GCS_Tokyo = 4301
GCS_Trinidad_1903 = 4302
GCS_TC_1948 = 4303
GCS_Voirol_1875 = 4304
GCS_Voirol_Unifie = 4305
GCS_Bern_1938 = 4306
GCS_Nord_Sahara_1959 = 4307
GCS_Stockholm_1938 = 4308
GCS_Yacare = 4309
GCS_Yoff = 4310
GCS_Zanderij = 4311
GCS_MGI = 4312
GCS_Belge_1972 = 4313
GCS_DHDN = 4314
GCS_Conakry_1905 = 4315
GCS_WGS_72 = 4322
GCS_WGS_72BE = 4324
GCS_WGS_84 = 4326
GCS_Bern_1898_Bern = 4801
GCS_Bogota_Bogota = 4802
GCS_Lisbon_Lisbon = 4803
GCS_Makassar_Jakarta = 4804
GCS_MGI_Ferro = 4805
GCS_Monte_Mario_Rome = 4806
GCS_NTF_Paris = 4807

```
GCS_Padang_Jakarta =      4808
GCS_Belge_1950_Brussels = 4809
GCS_Tananarive_Paris =    4810
GCS_Voirol_1875_Paris =   4811
GCS_Voirol_Unifie_Paris = 4812
GCS_Batavia_Jakarta =     4813
GCS_ATF_Paris =    4901
GCS_NDG_Paris =    4902
```

Ellipsoid-Only GCS:

Note: the numeric code is equal to the code of the correspoding EPSG ellipsoid, minus 3000.

```
GCSE_Airy1830 =    4001
GCSE_AiryModified1849 =   4002
GCSE_AustralianNationalSpheroid =       4003
GCSE_Bessel1841 =  4004
GCSE_BesselModified =     4005
GCSE_BesselNamibia =      4006
GCSE_Clarke1858 =  4007
GCSE_Clarke1866 =  4008
GCSE_Clarke1866Michigan = 4009
GCSE_Clarke1880_Benoit =  4010
GCSE_Clarke1880_IGN =     4011
GCSE_Clarke1880_RGS =     4012
GCSE_Clarke1880_Arc =     4013
GCSE_Clarke1880_SGA1922 = 4014
GCSE_Everest1830_1937Adjustment =       4015
GCSE_Everest1830_1967Definition =       4016
GCSE_Everest1830_1975Definition =       4017
GCSE_Everest1830Modified =       4018
GCSE_GRS1980 =     4019
GCSE_Helmert1906 = 4020
GCSE_IndonesianNationalSpheroid =       4021
GCSE_International1924 =  4022
GCSE_International1967 =  4023
GCSE_Krassowsky1940 =     4024
GCSE_NWL9D =       4025
GCSE_NWL10D =      4026
GCSE_Plessis1817 = 4027
GCSE_Struve1860 =  4028
GCSE_WarOffice =   4029
GCSE_WGS84 =       4030
GCSE_GEM10C =      4031
GCSE_OSU86F =      4032
GCSE_OSU91A =      4033
GCSE_Clarke1880 =  4034
GCSE_Sphere =      4035


+----------------------------------+
```

6.3.2.2 Geodetic Datum Codes

Note: these codes do not include the Prime Meridian; if possible use the GCS codes above if the datum and Prime Meridian are on the list. Also, as with the GCS codes, the codes beginning with DatumE_xxx refer only to the specified ellipsoid (xxx); if possible use instead the named datums beginning with Datum_xxx

Ranges:,

```
0 = undefined
[    1,  1000] = Obsolete EPSG/POSC Datum Codes
[ 1001,  5999] = Reserved by GeoTIFF
[ 6000, 6199]  = EPSG Datum Based on Ellipsoid only
[ 6200, 6999]  = EPSG Datum Based on EPSG Datum
[ 6322, 6327]  = WGS Datum
[ 6900, 6999]  = Archaic Datum
[ 7000, 32766] = Reserved by GeoTIFF
32767          = user-defined GCS
[32768, 65535] = Private User Implementations
```

Values:

```
Datum_Adindan =    6201
Datum_Australian_Geodetic_Datum_1966 =   6202
Datum_Australian_Geodetic_Datum_1984 =   6203
Datum_Ain_el_Abd_1970 =   6204
Datum_Afgooye =    6205
Datum_Agadez =     6206
Datum_Lisbon =     6207
Datum_Aratu =      6208
Datum_Arc_1950 =   6209
Datum_Arc_1960 =   6210
Datum_Batavia =    6211
Datum_Barbados =   6212
Datum_Beduaram =   6213
Datum_Beijing_1954 =      6214
Datum_Reseau_National_Belge_1950 =      6215
Datum_Bermuda_1957 =      6216
Datum_Bern_1898 =  6217
Datum_Bogota =     6218
Datum_Bukit_Rimpah =      6219
Datum_Camacupa =   6220
Datum_Campo_Inchauspe =   6221
Datum_Cape =       6222
Datum_Carthage =   6223
Datum_Chua =       6224
Datum_Corrego_Alegre =    6225
Datum_Cote_d_Ivoire =     6226
Datum_Deir_ez_Zor =       6227
Datum_Douala =     6228
Datum_Egypt_1907 = 6229
Datum_European_Datum_1950 =      6230
Datum_European_Datum_1987 =      6231
Datum_Fahud =      6232
Datum_Gandajika_1970 =    6233
Datum_Garoua =     6234
Datum_Guyane_Francaise =  6235
Datum_Hu_Tzu_Shan =       6236
Datum_Hungarian_Datum_1972 =     6237
```

Datum_Indonesian_Datum_1974 = 6238
Datum_Indian_1954 = 6239
Datum_Indian_1975 = 6240
Datum_Jamaica_1875 = 6241
Datum_Jamaica_1969 = 6242
Datum_Kalianpur = 6243
Datum_Kandawala = 6244
Datum_Kertau = 6245
Datum_Kuwait_Oil_Company = 6246
Datum_La_Canoa = 6247
Datum_Provisional_S_American_Datum_1956 = 6248
Datum_Lake = 6249
Datum_Leigon = 6250
Datum_Liberia_1964 = 6251
Datum_Lome = 6252
Datum_Luzon_1911 = 6253
Datum_Hito_XVIII_1963 = 6254
Datum_Herat_North = 6255
Datum_Mahe_1971 = 6256
Datum_Makassar = 6257
Datum_European_Reference_System_1989 = 6258
Datum_Malongo_1987 = 6259
Datum_Manoca = 6260
Datum_Merchich = 6261
Datum_Massawa = 6262
Datum_Minna = 6263
Datum_Mhast = 6264
Datum_Monte_Mario = 6265
Datum_M_poraloko = 6266
Datum_North_American_Datum_1927 = 6267
Datum_NAD_Michigan = 6268
Datum_North_American_Datum_1983 = 6269
Datum_Nahrwan_1967 = 6270
Datum_Naparima_1972 = 6271
Datum_New_Zealand_Geodetic_Datum_1949 = 6272
Datum_NGO_1948 = 6273
Datum_Datum_73 = 6274
Datum_Nouvelle_Triangulation_Francaise = 6275
Datum_NSWC_9Z_2 = 6276
Datum_OSGB_1936 = 6277
Datum_OSGB_1970_SN = 6278
Datum_OS_SN_1980 = 6279
Datum_Padang_1884 = 6280
Datum_Palestine_1923 = 6281
Datum_Pointe_Noire = 6282
Datum_Geocentric_Datum_of_Australia_1994 = 6283
Datum_Pulkovo_1942 = 6284
Datum_Qatar = 6285
Datum_Qatar_1948 = 6286
Datum_Qornoq = 6287
Datum_Loma_Quintana = 6288
Datum_Amersfoort = 6289
Datum_RT38 = 6290
Datum_South_American_Datum_1969 = 6291
Datum_Sapper_Hill_1943 = 6292
Datum_Schwarzeck = 6293
Datum_Segora = 6294

```
Datum_Serindung =  6295
Datum_Sudan =      6296
Datum_Tananarive_1925 =   6297
Datum_Timbalai_1948 =     6298
Datum_TM65 =       6299
Datum_TM75 =       6300
Datum_Tokyo =      6301
Datum_Trinidad_1903 =     6302
Datum_Trucial_Coast_1948 =       6303
Datum_Voirol_1875 =       6304
Datum_Voirol_Unifie_1960 =       6305
Datum_Bern_1938 =  6306
Datum_Nord_Sahara_1959 =  6307
Datum_Stockholm_1938 =    6308
Datum_Yacare =     6309
Datum_Yoff =       6310
Datum_Zanderij =   6311
Datum_Militar_Geographische_Institut =  6312
Datum_Reseau_National_Belge_1972 =      6313
Datum_Deutsche_Hauptdreiecksnetz =      6314
Datum_Conakry_1905 =      6315
Datum_WGS72 =      6322
Datum_WGS72_Transit_Broadcast_Ephemeris =      6324
Datum_WGS84 =      6326
Datum_Ancienne_Triangulation_Francaise = 6901
Datum_Nord_de_Guerre =    6902
```

Ellipsoid-Only Datum:

Note: the numeric code is equal to the corresponding ellipsoid code, minus 1000.

```
DatumE_Airy1830 =  6001
DatumE_AiryModified1849 = 6002
DatumE_AustralianNationalSpheroid =     6003
DatumE_Bessel1841 =       6004
DatumE_BesselModified =   6005
DatumE_BesselNamibia =    6006
DatumE_Clarke1858 =       6007
DatumE_Clarke1866 =       6008
DatumE_Clarke1866Michigan =      6009
DatumE_Clarke1880_Benoit =       6010
DatumE_Clarke1880_IGN =   6011
DatumE_Clarke1880_RGS =   6012
DatumE_Clarke1880_Arc =   6013
DatumE_Clarke1880_SGA1922 =      6014
DatumE_Everest1830_1937Adjustment =     6015
DatumE_Everest1830_1967Definition =     6016
DatumE_Everest1830_1975Definition =     6017
DatumE_Everest1830Modified =     6018
DatumE_GRS1980 =   6019
DatumE_Helmert1906 =      6020
DatumE_IndonesianNationalSpheroid =     6021
DatumE_International1924 =       6022
DatumE_International1967 =       6023
DatumE_Krassowsky1960 =   6024
DatumE_NWL9D =     6025
```

```
DatumE_NWL10D =    6026
DatumE_Plessis1817 =      6027
DatumE_Struve1860 =       6028
DatumE_WarOffice = 6029
DatumE_WGS84 =     6030
DatumE_GEM10C =    6031
DatumE_OSU86F =    6032
DatumE_OSU91A =    6033
DatumE_Clarke1880 =       6034
DatumE_Sphere =    6035

+-----------------------------------+
```

6.3.2.3 Ellipsoid Codes

```
Ranges:

   0 = undefined
   [    1, 1000]  = Obsolete EPSG/POSC Ellipsoid codes
   [1001,  6999]  = Reserved by GeoTIFF
   [7000,  7999]  = EPSG Ellipsoid codes
   [8000, 32766]  = Reserved by GeoTIFF
   32767          = user-defined
   [32768, 65535] = Private User Implementations

Values:

   Ellipse_Airy_1830 =       7001
   Ellipse_Airy_Modified_1849 =     7002
   Ellipse_Australian_National_Spheroid =   7003
   Ellipse_Bessel_1841 =     7004
   Ellipse_Bessel_Modified = 7005
   Ellipse_Bessel_Namibia =  7006
   Ellipse_Clarke_1858 =     7007
   Ellipse_Clarke_1866 =     7008
   Ellipse_Clarke_1866_Michigan =   7009
   Ellipse_Clarke_1880_Benoit =     7010
   Ellipse_Clarke_1880_IGN = 7011
   Ellipse_Clarke_1880_RGS = 7012
   Ellipse_Clarke_1880_Arc = 7013
   Ellipse_Clarke_1880_SGA_1922 =   7014
   Ellipse_Everest_1830_1937_Adjustment =   7015
   Ellipse_Everest_1830_1967_Definition =   7016
   Ellipse_Everest_1830_1975_Definition =   7017
   Ellipse_Everest_1830_Modified =  7018
   Ellipse_GRS_1980 = 7019
   Ellipse_Helmert_1906 =    7020
   Ellipse_Indonesian_National_Spheroid =   7021
   Ellipse_International_1924 =     7022
   Ellipse_International_1967 =     7023
   Ellipse_Krassowsky_1940 = 7024
   Ellipse_NWL_9D =   7025
   Ellipse_NWL_10D =  7026
   Ellipse_Plessis_1817 =    7027
   Ellipse_Struve_1860 =     7028
   Ellipse_War_Office =      7029
```

```
Ellipse_WGS_84 =   7030
Ellipse_GEM_10C =  7031
Ellipse_OSU86F =   7032
Ellipse_OSU91A =   7033
Ellipse_Clarke_1880 =     7034
Ellipse_Sphere =   7035
```

+----------------------------------+

6.3.2.4 Prime Meridian Codes

Ranges:

```
0 = undefined
[    1,   100] = Obsolete EPSG/POSC Prime Meridian codes
[  101,  7999] = Reserved by GeoTIFF
[ 8000,  8999] = EPSG Prime Meridian Codes
[ 9000, 32766] = Reserved by GeoTIFF
32767          = user-defined
[32768, 65535] = Private User Implementations
```

Values:

```
PM_Greenwich =     8901
PM_Lisbon = 8902
PM_Paris =  8903
PM_Bogota = 8904
PM_Madrid = 8905
PM_Rome =   8906
PM_Bern =   8907
PM_Jakarta =       8908
PM_Ferro =  8909
PM_Brussels =      8910
PM_Stockholm =     8911
```

+----------------------------------+

6.3.3 Projected CS Codes

+----------------------------------+

6.3.3.1 Projected CS Type Codes

Ranges:

```
[    1,   1000]  = Obsolete EPSG/POSC Projection System Codes
[20000,  32760]  = EPSG Projection System codes
32767            = user-defined
[32768,  65535]  = Private User Implementations
```

Special Ranges:

```
1. For PCS utilising GeogCS with code in range 4201 through 4321
(i.e. geodetic datum code 6201 through 6319): As far as is possible
 the PCS code will be of the format gggzz where ggg is (geodetic
datum code -2000) and zz is zone.

2. For PCS utilising GeogCS with code out of range 4201 through 4321
(i.e. geodetic datum code 6201 through 6319). PCS code 20xxx where
xxx is a sequential number.

3. Other:
   WGS72 / UTM northern hemisphere: 322zz where zz is UTM zone number
   WGS72 / UTM southern hemisphere: 323zz where zz is UTM zone number
   WGS72BE / UTM northern hemisphere: 324zz where zz is UTM zone number
   WGS72BE / UTM southern hemisphere: 325zz where zz is UTM zone number
   WGS84 / UTM northern hemisphere: 326zz where zz is UTM zone number
   WGS84 / UTM southern hemisphere: 327zz where zz is UTM zone number
   US State Plane (NAD27):   267xx/320xx
   US State Plane (NAD83):   269xx/321xx


Values:

   PCS_Adindan_UTM_zone_37N =       20137
   PCS_Adindan_UTM_zone_38N =       20138
   PCS_AGD66_AMG_zone_48 =   20248
   PCS_AGD66_AMG_zone_49 =   20249
   PCS_AGD66_AMG_zone_50 =   20250
   PCS_AGD66_AMG_zone_51 =   20251
   PCS_AGD66_AMG_zone_52 =   20252
   PCS_AGD66_AMG_zone_53 =   20253
   PCS_AGD66_AMG_zone_54 =   20254
   PCS_AGD66_AMG_zone_55 =   20255
   PCS_AGD66_AMG_zone_56 =   20256
   PCS_AGD66_AMG_zone_57 =   20257
   PCS_AGD66_AMG_zone_58 =   20258
   PCS_AGD84_AMG_zone_48 =   20348
   PCS_AGD84_AMG_zone_49 =   20349
   PCS_AGD84_AMG_zone_50 =   20350
   PCS_AGD84_AMG_zone_51 =   20351
   PCS_AGD84_AMG_zone_52 =   20352
   PCS_AGD84_AMG_zone_53 =   20353
   PCS_AGD84_AMG_zone_54 =   20354
   PCS_AGD84_AMG_zone_55 =   20355
   PCS_AGD84_AMG_zone_56 =   20356
   PCS_AGD84_AMG_zone_57 =   20357
   PCS_AGD84_AMG_zone_58 =   20358
   PCS_Ain_el_Abd_UTM_zone_37N =    20437
   PCS_Ain_el_Abd_UTM_zone_38N =    20438
   PCS_Ain_el_Abd_UTM_zone_39N =    20439
   PCS_Ain_el_Abd_Bahrain_Grid =    20499
   PCS_Afgooye_UTM_zone_38N =       20538
   PCS_Afgooye_UTM_zone_39N =       20539
   PCS_Lisbon_Portugese_Grid =      20700
   PCS_Aratu_UTM_zone_22S =  20822
   PCS_Aratu_UTM_zone_23S =  20823
   PCS_Aratu_UTM_zone_24S =  20824
```

```
PCS_Arc_1950_Lo13 =       20973
PCS_Arc_1950_Lo15 =       20975
PCS_Arc_1950_Lo17 =       20977
PCS_Arc_1950_Lo19 =       20979
PCS_Arc_1950_Lo21 =       20981
PCS_Arc_1950_Lo23 =       20983
PCS_Arc_1950_Lo25 =       20985
PCS_Arc_1950_Lo27 =       20987
PCS_Arc_1950_Lo29 =       20989
PCS_Arc_1950_Lo31 =       20991
PCS_Arc_1950_Lo33 =       20993
PCS_Arc_1950_Lo35 =       20995
PCS_Batavia_NEIEZ =       21100
PCS_Batavia_UTM_zone_48S =       21148
PCS_Batavia_UTM_zone_49S =       21149
PCS_Batavia_UTM_zone_50S =       21150
PCS_Beijing_Gauss_zone_13 =      21413
PCS_Beijing_Gauss_zone_14 =      21414
PCS_Beijing_Gauss_zone_15 =      21415
PCS_Beijing_Gauss_zone_16 =      21416
PCS_Beijing_Gauss_zone_17 =      21417
PCS_Beijing_Gauss_zone_18 =      21418
PCS_Beijing_Gauss_zone_19 =      21419
PCS_Beijing_Gauss_zone_20 =      21420
PCS_Beijing_Gauss_zone_21 =      21421
PCS_Beijing_Gauss_zone_22 =      21422
PCS_Beijing_Gauss_zone_23 =      21423
PCS_Beijing_Gauss_13N =   21473
PCS_Beijing_Gauss_14N =   21474
PCS_Beijing_Gauss_15N =   21475
PCS_Beijing_Gauss_16N =   21476
PCS_Beijing_Gauss_17N =   21477
PCS_Beijing_Gauss_18N =   21478
PCS_Beijing_Gauss_19N =   21479
PCS_Beijing_Gauss_20N =   21480
PCS_Beijing_Gauss_21N =   21481
PCS_Beijing_Gauss_22N =   21482
PCS_Beijing_Gauss_23N =   21483
PCS_Belge_Lambert_50 =    21500
PCS_Bern_1898_Swiss_Old = 21790
PCS_Bogota_UTM_zone_17N = 21817
PCS_Bogota_UTM_zone_18N = 21818
PCS_Bogota_Colombia_3W =  21891
PCS_Bogota_Colombia_Bogota =     21892
PCS_Bogota_Colombia_3E =  21893
PCS_Bogota_Colombia_6E =  21894
PCS_Camacupa_UTM_32S =    22032
PCS_Camacupa_UTM_33S =    22033
PCS_C_Inchauspe_Argentina_1 =    22191
PCS_C_Inchauspe_Argentina_2 =    22192
PCS_C_Inchauspe_Argentina_3 =    22193
PCS_C_Inchauspe_Argentina_4 =    22194
PCS_C_Inchauspe_Argentina_5 =    22195
PCS_C_Inchauspe_Argentina_6 =    22196
PCS_C_Inchauspe_Argentina_7 =    22197
PCS_Carthage_UTM_zone_32N =      22332
PCS_Carthage_Nord_Tunisie =      22391
```

```
PCS_Carthage_Sud_Tunisie =       22392
PCS_Corrego_Alegre_UTM_23S =     22523
PCS_Corrego_Alegre_UTM_24S =     22524
PCS_Douala_UTM_zone_32N = 22832
PCS_Egypt_1907_Red_Belt = 22992
PCS_Egypt_1907_Purple_Belt =     22993
PCS_Egypt_1907_Ext_Purple =      22994
PCS_ED50_UTM_zone_28N =   23028
PCS_ED50_UTM_zone_29N =   23029
PCS_ED50_UTM_zone_30N =   23030
PCS_ED50_UTM_zone_31N =   23031
PCS_ED50_UTM_zone_32N =   23032
PCS_ED50_UTM_zone_33N =   23033
PCS_ED50_UTM_zone_34N =   23034
PCS_ED50_UTM_zone_35N =   23035
PCS_ED50_UTM_zone_36N =   23036
PCS_ED50_UTM_zone_37N =   23037
PCS_ED50_UTM_zone_38N =   23038
PCS_Fahud_UTM_zone_39N =  23239
PCS_Fahud_UTM_zone_40N =  23240
PCS_Garoua_UTM_zone_33N = 23433
PCS_ID74_UTM_zone_46N =   23846
PCS_ID74_UTM_zone_47N =   23847
PCS_ID74_UTM_zone_48N =   23848
PCS_ID74_UTM_zone_49N =   23849
PCS_ID74_UTM_zone_50N =   23850
PCS_ID74_UTM_zone_51N =   23851
PCS_ID74_UTM_zone_52N =   23852
PCS_ID74_UTM_zone_53N =   23853
PCS_ID74_UTM_zone_46S =   23886
PCS_ID74_UTM_zone_47S =   23887
PCS_ID74_UTM_zone_48S =   23888
PCS_ID74_UTM_zone_49S =   23889
PCS_ID74_UTM_zone_50S =   23890
PCS_ID74_UTM_zone_51S =   23891
PCS_ID74_UTM_zone_52S =   23892
PCS_ID74_UTM_zone_53S =   23893
PCS_ID74_UTM_zone_54S =   23894
PCS_Indian_1954_UTM_47N = 23947
PCS_Indian_1954_UTM_48N = 23948
PCS_Indian_1975_UTM_47N = 24047
PCS_Indian_1975_UTM_48N = 24048
PCS_Jamaica_1875_Old_Grid =      24100
PCS_JAD69_Jamaica_Grid =  24200
PCS_Kalianpur_India_0 =   24370
PCS_Kalianpur_India_I =   24371
PCS_Kalianpur_India_IIa = 24372
PCS_Kalianpur_India_IIIa =       24373
PCS_Kalianpur_India_IVa = 24374
PCS_Kalianpur_India_IIb = 24382
PCS_Kalianpur_India_IIIb =       24383
PCS_Kalianpur_India_IVb = 24384
PCS_Kertau_Singapore_Grid =      24500
PCS_Kertau_UTM_zone_47N = 24547
PCS_Kertau_UTM_zone_48N = 24548
PCS_La_Canoa_UTM_zone_20N =      24720
PCS_La_Canoa_UTM_zone_21N =      24721
```

PCS_PSAD56_UTM_zone_18N = 24818
PCS_PSAD56_UTM_zone_19N = 24819
PCS_PSAD56_UTM_zone_20N = 24820
PCS_PSAD56_UTM_zone_21N = 24821
PCS_PSAD56_UTM_zone_17S = 24877
PCS_PSAD56_UTM_zone_18S = 24878
PCS_PSAD56_UTM_zone_19S = 24879
PCS_PSAD56_UTM_zone_20S = 24880
PCS_PSAD56_Peru_west_zone = 24891
PCS_PSAD56_Peru_central = 24892
PCS_PSAD56_Peru_east_zone = 24893
PCS_Leigon_Ghana_Grid = 25000
PCS_Lome_UTM_zone_31N = 25231
PCS_Luzon_Philippines_I = 25391
PCS_Luzon_Philippines_II = 25392
PCS_Luzon_Philippines_III = 25393
PCS_Luzon_Philippines_IV = 25394
PCS_Luzon_Philippines_V = 25395
PCS_Makassar_NEIEZ = 25700
PCS_Malongo_1987_UTM_32S = 25932
PCS_Merchich_Nord_Maroc = 26191
PCS_Merchich_Sud_Maroc = 26192
PCS_Merchich_Sahara = 26193
PCS_Massawa_UTM_zone_37N = 26237
PCS_Minna_UTM_zone_31N = 26331
PCS_Minna_UTM_zone_32N = 26332
PCS_Minna_Nigeria_West = 26391
PCS_Minna_Nigeria_Mid_Belt = 26392
PCS_Minna_Nigeria_East = 26393
PCS_Mhast_UTM_zone_32S = 26432
PCS_Monte_Mario_Italy_1 = 26591
PCS_Monte_Mario_Italy_2 = 26592
PCS_M_poraloko_UTM_32N = 26632
PCS_M_poraloko_UTM_32S = 26692
PCS_NAD27_UTM_zone_3N = 26703
PCS_NAD27_UTM_zone_4N = 26704
PCS_NAD27_UTM_zone_5N = 26705
PCS_NAD27_UTM_zone_6N = 26706
PCS_NAD27_UTM_zone_7N = 26707
PCS_NAD27_UTM_zone_8N = 26708
PCS_NAD27_UTM_zone_9N = 26709
PCS_NAD27_UTM_zone_10N = 26710
PCS_NAD27_UTM_zone_11N = 26711
PCS_NAD27_UTM_zone_12N = 26712
PCS_NAD27_UTM_zone_13N = 26713
PCS_NAD27_UTM_zone_14N = 26714
PCS_NAD27_UTM_zone_15N = 26715
PCS_NAD27_UTM_zone_16N = 26716
PCS_NAD27_UTM_zone_17N = 26717
PCS_NAD27_UTM_zone_18N = 26718
PCS_NAD27_UTM_zone_19N = 26719
PCS_NAD27_UTM_zone_20N = 26720
PCS_NAD27_UTM_zone_21N = 26721
PCS_NAD27_UTM_zone_22N = 26722
PCS_NAD27_Alabama_East = 26729
PCS_NAD27_Alabama_West = 26730
PCS_NAD27_Alaska_zone_1 = 26731

```
PCS_NAD27_Alaska_zone_2 = 26732
PCS_NAD27_Alaska_zone_3 = 26733
PCS_NAD27_Alaska_zone_4 = 26734
PCS_NAD27_Alaska_zone_5 = 26735
PCS_NAD27_Alaska_zone_6 = 26736
PCS_NAD27_Alaska_zone_7 = 26737
PCS_NAD27_Alaska_zone_8 = 26738
PCS_NAD27_Alaska_zone_9 = 26739
PCS_NAD27_Alaska_zone_10 =       26740
PCS_NAD27_California_I =  26741
PCS_NAD27_California_II = 26742
PCS_NAD27_California_III =       26743
PCS_NAD27_California_IV = 26744
PCS_NAD27_California_V =  26745
PCS_NAD27_California_VI = 26746
PCS_NAD27_California_VII =       26747
PCS_NAD27_Arizona_East =  26748
PCS_NAD27_Arizona_Central =      26749
PCS_NAD27_Arizona_West =  26750
PCS_NAD27_Arkansas_North =       26751
PCS_NAD27_Arkansas_South =       26752
PCS_NAD27_Colorado_North =       26753
PCS_NAD27_Colorado_Central =     26754
PCS_NAD27_Colorado_South =       26755
PCS_NAD27_Connecticut =   26756
PCS_NAD27_Delaware =      26757
PCS_NAD27_Florida_East =  26758
PCS_NAD27_Florida_West =  26759
PCS_NAD27_Florida_North = 26760
PCS_NAD27_Hawaii_zone_1 = 26761
PCS_NAD27_Hawaii_zone_2 = 26762
PCS_NAD27_Hawaii_zone_3 = 26763
PCS_NAD27_Hawaii_zone_4 = 26764
PCS_NAD27_Hawaii_zone_5 = 26765
PCS_NAD27_Georgia_East =  26766
PCS_NAD27_Georgia_West =  26767
PCS_NAD27_Idaho_East =    26768
PCS_NAD27_Idaho_Central = 26769
PCS_NAD27_Idaho_West =    26770
PCS_NAD27_Illinois_East = 26771
PCS_NAD27_Illinois_West = 26772
PCS_NAD27_Indiana_East =  26773
PCS_NAD27_BLM_14N_feet =  26774
PCS_NAD27_Indiana_West =  26774
PCS_NAD27_BLM_15N_feet =  26775
PCS_NAD27_Iowa_North =    26775
PCS_NAD27_BLM_16N_feet =  26776
PCS_NAD27_Iowa_South =    26776
PCS_NAD27_BLM_17N_feet =  26777
PCS_NAD27_Kansas_North =  26777
PCS_NAD27_Kansas_South =  26778
PCS_NAD27_Kentucky_North =       26779
PCS_NAD27_Kentucky_South =       26780
PCS_NAD27_Louisiana_North =      26781
PCS_NAD27_Louisiana_South =      26782
PCS_NAD27_Maine_East =    26783
PCS_NAD27_Maine_West =    26784
```

PCS_NAD27_Maryland = 26785
PCS_NAD27_Massachusetts = 26786
PCS_NAD27_Massachusetts_Is = 26787
PCS_NAD27_Michigan_North = 26788
PCS_NAD27_Michigan_Central = 26789
PCS_NAD27_Michigan_South = 26790
PCS_NAD27_Minnesota_North = 26791
PCS_NAD27_Minnesota_Cent = 26792
PCS_NAD27_Minnesota_South = 26793
PCS_NAD27_Mississippi_East = 26794
PCS_NAD27_Mississippi_West = 26795
PCS_NAD27_Missouri_East = 26796
PCS_NAD27_Missouri_Central = 26797
PCS_NAD27_Missouri_West = 26798
PCS_NAD_Michigan_Michigan_East = 26801
PCS_NAD_Michigan_Michigan_Old_Central = 26802
PCS_NAD_Michigan_Michigan_West = 26803
PCS_NAD83_UTM_zone_3N = 26903
PCS_NAD83_UTM_zone_4N = 26904
PCS_NAD83_UTM_zone_5N = 26905
PCS_NAD83_UTM_zone_6N = 26906
PCS_NAD83_UTM_zone_7N = 26907
PCS_NAD83_UTM_zone_8N = 26908
PCS_NAD83_UTM_zone_9N = 26909
PCS_NAD83_UTM_zone_10N = 26910
PCS_NAD83_UTM_zone_11N = 26911
PCS_NAD83_UTM_zone_12N = 26912
PCS_NAD83_UTM_zone_13N = 26913
PCS_NAD83_UTM_zone_14N = 26914
PCS_NAD83_UTM_zone_15N = 26915
PCS_NAD83_UTM_zone_16N = 26916
PCS_NAD83_UTM_zone_17N = 26917
PCS_NAD83_UTM_zone_18N = 26918
PCS_NAD83_UTM_zone_19N = 26919
PCS_NAD83_UTM_zone_20N = 26920
PCS_NAD83_UTM_zone_21N = 26921
PCS_NAD83_UTM_zone_22N = 26922
PCS_NAD83_UTM_zone_23N = 26923
PCS_NAD83_Alabama_East = 26929
PCS_NAD83_Alabama_West = 26930
PCS_NAD83_Alaska_zone_1 = 26931
PCS_NAD83_Alaska_zone_2 = 26932
PCS_NAD83_Alaska_zone_3 = 26933
PCS_NAD83_Alaska_zone_4 = 26934
PCS_NAD83_Alaska_zone_5 = 26935
PCS_NAD83_Alaska_zone_6 = 26936
PCS_NAD83_Alaska_zone_7 = 26937
PCS_NAD83_Alaska_zone_8 = 26938
PCS_NAD83_Alaska_zone_9 = 26939
PCS_NAD83_Alaska_zone_10 = 26940
PCS_NAD83_California_1 = 26941
PCS_NAD83_California_2 = 26942
PCS_NAD83_California_3 = 26943
PCS_NAD83_California_4 = 26944
PCS_NAD83_California_5 = 26945
PCS_NAD83_California_6 = 26946
PCS_NAD83_Arizona_East = 26948

```
PCS_NAD83_Arizona_Central =      26949
PCS_NAD83_Arizona_West =  26950
PCS_NAD83_Arkansas_North =       26951
PCS_NAD83_Arkansas_South =       26952
PCS_NAD83_Colorado_North =       26953
PCS_NAD83_Colorado_Central =     26954
PCS_NAD83_Colorado_South =       26955
PCS_NAD83_Connecticut =   26956
PCS_NAD83_Delaware =      26957
PCS_NAD83_Florida_East =  26958
PCS_NAD83_Florida_West =  26959
PCS_NAD83_Florida_North = 26960
PCS_NAD83_Hawaii_zone_1 = 26961
PCS_NAD83_Hawaii_zone_2 = 26962
PCS_NAD83_Hawaii_zone_3 = 26963
PCS_NAD83_Hawaii_zone_4 = 26964
PCS_NAD83_Hawaii_zone_5 = 26965
PCS_NAD83_Georgia_East =  26966
PCS_NAD83_Georgia_West =  26967
PCS_NAD83_Idaho_East =    26968
PCS_NAD83_Idaho_Central = 26969
PCS_NAD83_Idaho_West =    26970
PCS_NAD83_Illinois_East = 26971
PCS_NAD83_Illinois_West = 26972
PCS_NAD83_Indiana_East =  26973
PCS_NAD83_Indiana_West =  26974
PCS_NAD83_Iowa_North =    26975
PCS_NAD83_Iowa_South =    26976
PCS_NAD83_Kansas_North =  26977
PCS_NAD83_Kansas_South =  26978
PCS_NAD83_Kentucky_North =       26979
PCS_NAD83_Kentucky_South =       26980
PCS_NAD83_Louisiana_North =      26981
PCS_NAD83_Louisiana_South =      26982
PCS_NAD83_Maine_East =    26983
PCS_NAD83_Maine_West =    26984
PCS_NAD83_Maryland =      26985
PCS_NAD83_Massachusetts = 26986
PCS_NAD83_Massachusetts_Is =     26987
PCS_NAD83_Michigan_North =       26988
PCS_NAD83_Michigan_Central =     26989
PCS_NAD83_Michigan_South =       26990
PCS_NAD83_Minnesota_North =      26991
PCS_NAD83_Minnesota_Cent =       26992
PCS_NAD83_Minnesota_South =      26993
PCS_NAD83_Mississippi_East =     26994
PCS_NAD83_Mississippi_West =     26995
PCS_NAD83_Missouri_East = 26996
PCS_NAD83_Missouri_Central =     26997
PCS_NAD83_Missouri_West = 26998
PCS_Nahrwan_1967_UTM_38N =       27038
PCS_Nahrwan_1967_UTM_39N =       27039
PCS_Nahrwan_1967_UTM_40N =       27040
PCS_Naparima_UTM_20N =    27120
PCS_GD49_NZ_Map_Grid =    27200
PCS_GD49_North_Island_Grid =     27291
PCS_GD49_South_Island_Grid =     27292
```

PCS_Datum_73_UTM_zone_29N = 27429
PCS_ATF_Nord_de_Guerre = 27500
PCS_NTF_France_I = 27581
PCS_NTF_France_II = 27582
PCS_NTF_France_III = 27583
PCS_NTF_Nord_France = 27591
PCS_NTF_Centre_France = 27592
PCS_NTF_Sud_France = 27593
PCS_British_National_Grid = 27700
PCS_Point_Noire_UTM_32S = 28232
PCS_GDA94_MGA_zone_48 = 28348
PCS_GDA94_MGA_zone_49 = 28349
PCS_GDA94_MGA_zone_50 = 28350
PCS_GDA94_MGA_zone_51 = 28351
PCS_GDA94_MGA_zone_52 = 28352
PCS_GDA94_MGA_zone_53 = 28353
PCS_GDA94_MGA_zone_54 = 28354
PCS_GDA94_MGA_zone_55 = 28355
PCS_GDA94_MGA_zone_56 = 28356
PCS_GDA94_MGA_zone_57 = 28357
PCS_GDA94_MGA_zone_58 = 28358
PCS_Pulkovo_Gauss_zone_4 = 28404
PCS_Pulkovo_Gauss_zone_5 = 28405
PCS_Pulkovo_Gauss_zone_6 = 28406
PCS_Pulkovo_Gauss_zone_7 = 28407
PCS_Pulkovo_Gauss_zone_8 = 28408
PCS_Pulkovo_Gauss_zone_9 = 28409
PCS_Pulkovo_Gauss_zone_10 = 28410
PCS_Pulkovo_Gauss_zone_11 = 28411
PCS_Pulkovo_Gauss_zone_12 = 28412
PCS_Pulkovo_Gauss_zone_13 = 28413
PCS_Pulkovo_Gauss_zone_14 = 28414
PCS_Pulkovo_Gauss_zone_15 = 28415
PCS_Pulkovo_Gauss_zone_16 = 28416
PCS_Pulkovo_Gauss_zone_17 = 28417
PCS_Pulkovo_Gauss_zone_18 = 28418
PCS_Pulkovo_Gauss_zone_19 = 28419
PCS_Pulkovo_Gauss_zone_20 = 28420
PCS_Pulkovo_Gauss_zone_21 = 28421
PCS_Pulkovo_Gauss_zone_22 = 28422
PCS_Pulkovo_Gauss_zone_23 = 28423
PCS_Pulkovo_Gauss_zone_24 = 28424
PCS_Pulkovo_Gauss_zone_25 = 28425
PCS_Pulkovo_Gauss_zone_26 = 28426
PCS_Pulkovo_Gauss_zone_27 = 28427
PCS_Pulkovo_Gauss_zone_28 = 28428
PCS_Pulkovo_Gauss_zone_29 = 28429
PCS_Pulkovo_Gauss_zone_30 = 28430
PCS_Pulkovo_Gauss_zone_31 = 28431
PCS_Pulkovo_Gauss_zone_32 = 28432
PCS_Pulkovo_Gauss_4N = 28464
PCS_Pulkovo_Gauss_5N = 28465
PCS_Pulkovo_Gauss_6N = 28466
PCS_Pulkovo_Gauss_7N = 28467
PCS_Pulkovo_Gauss_8N = 28468
PCS_Pulkovo_Gauss_9N = 28469
PCS_Pulkovo_Gauss_10N = 28470

```
PCS_Pulkovo_Gauss_11N =   28471
PCS_Pulkovo_Gauss_12N =   28472
PCS_Pulkovo_Gauss_13N =   28473
PCS_Pulkovo_Gauss_14N =   28474
PCS_Pulkovo_Gauss_15N =   28475
PCS_Pulkovo_Gauss_16N =   28476
PCS_Pulkovo_Gauss_17N =   28477
PCS_Pulkovo_Gauss_18N =   28478
PCS_Pulkovo_Gauss_19N =   28479
PCS_Pulkovo_Gauss_20N =   28480
PCS_Pulkovo_Gauss_21N =   28481
PCS_Pulkovo_Gauss_22N =   28482
PCS_Pulkovo_Gauss_23N =   28483
PCS_Pulkovo_Gauss_24N =   28484
PCS_Pulkovo_Gauss_25N =   28485
PCS_Pulkovo_Gauss_26N =   28486
PCS_Pulkovo_Gauss_27N =   28487
PCS_Pulkovo_Gauss_28N =   28488
PCS_Pulkovo_Gauss_29N =   28489
PCS_Pulkovo_Gauss_30N =   28490
PCS_Pulkovo_Gauss_31N =   28491
PCS_Pulkovo_Gauss_32N =   28492
PCS_Qatar_National_Grid = 28600
PCS_RD_Netherlands_Old =  28991
PCS_RD_Netherlands_New =  28992
PCS_SAD69_UTM_zone_18N =  29118
PCS_SAD69_UTM_zone_19N =  29119
PCS_SAD69_UTM_zone_20N =  29120
PCS_SAD69_UTM_zone_21N =  29121
PCS_SAD69_UTM_zone_22N =  29122
PCS_SAD69_UTM_zone_17S =  29177
PCS_SAD69_UTM_zone_18S =  29178
PCS_SAD69_UTM_zone_19S =  29179
PCS_SAD69_UTM_zone_20S =  29180
PCS_SAD69_UTM_zone_21S =  29181
PCS_SAD69_UTM_zone_22S =  29182
PCS_SAD69_UTM_zone_23S =  29183
PCS_SAD69_UTM_zone_24S =  29184
PCS_SAD69_UTM_zone_25S =  29185
PCS_Sapper_Hill_UTM_20S = 29220
PCS_Sapper_Hill_UTM_21S = 29221
PCS_Schwarzeck_UTM_33S =  29333
PCS_Sudan_UTM_zone_35N =  29635
PCS_Sudan_UTM_zone_36N =  29636
PCS_Tananarive_Laborde =  29700
PCS_Tananarive_UTM_38S =  29738
PCS_Tananarive_UTM_39S =  29739
PCS_Timbalai_1948_Borneo =       29800
PCS_Timbalai_1948_UTM_49N =      29849
PCS_Timbalai_1948_UTM_50N =      29850
PCS_TM65_Irish_Nat_Grid = 29900
PCS_Trinidad_1903_Trinidad =     30200
PCS_TC_1948_UTM_zone_39N =       30339
PCS_TC_1948_UTM_zone_40N =       30340
PCS_Voirol_N_Algerie_ancien =    30491
PCS_Voirol_S_Algerie_ancien =    30492
PCS_Voirol_Unifie_N_Algerie =    30591
```

```
PCS_Voirol_Unifie_S_Algerie =    30592
PCS_Bern_1938_Swiss_New = 30600
PCS_Nord_Sahara_UTM_29N = 30729
PCS_Nord_Sahara_UTM_30N = 30730
PCS_Nord_Sahara_UTM_31N = 30731
PCS_Nord_Sahara_UTM_32N = 30732
PCS_Yoff_UTM_zone_28N =   31028
PCS_Zanderij_UTM_zone_21N =      31121
PCS_MGI_Austria_West =    31291
PCS_MGI_Austria_Central = 31292
PCS_MGI_Austria_East =    31293
PCS_Belge_Lambert_72 =    31300
PCS_DHDN_Germany_zone_1 = 31491
PCS_DHDN_Germany_zone_2 = 31492
PCS_DHDN_Germany_zone_3 = 31493
PCS_DHDN_Germany_zone_4 = 31494
PCS_DHDN_Germany_zone_5 = 31495
PCS_NAD27_Montana_North = 32001
PCS_NAD27_Montana_Central =      32002
PCS_NAD27_Montana_South = 32003
PCS_NAD27_Nebraska_North =       32005
PCS_NAD27_Nebraska_South =       32006
PCS_NAD27_Nevada_East =   32007
PCS_NAD27_Nevada_Central =       32008
PCS_NAD27_Nevada_West =   32009
PCS_NAD27_New_Hampshire = 32010
PCS_NAD27_New_Jersey =    32011
PCS_NAD27_New_Mexico_East =      32012
PCS_NAD27_New_Mexico_Cent =      32013
PCS_NAD27_New_Mexico_West =      32014
PCS_NAD27_New_York_East = 32015
PCS_NAD27_New_York_Central =     32016
PCS_NAD27_New_York_West = 32017
PCS_NAD27_New_York_Long_Is =     32018
PCS_NAD27_North_Carolina =       32019
PCS_NAD27_North_Dakota_N =       32020
PCS_NAD27_North_Dakota_S =       32021
PCS_NAD27_Ohio_North =    32022
PCS_NAD27_Ohio_South =    32023
PCS_NAD27_Oklahoma_North =       32024
PCS_NAD27_Oklahoma_South =       32025
PCS_NAD27_Oregon_North =  32026
PCS_NAD27_Oregon_South =  32027
PCS_NAD27_Pennsylvania_N =       32028
PCS_NAD27_Pennsylvania_S =       32029
PCS_NAD27_Rhode_Island =  32030
PCS_NAD27_South_Carolina_N =     32031
PCS_NAD27_South_Carolina_S =     32033
PCS_NAD27_South_Dakota_N =       32034
PCS_NAD27_South_Dakota_S =       32035
PCS_NAD27_Tennessee =     32036
PCS_NAD27_Texas_North =   32037
PCS_NAD27_Texas_North_Cen =      32038
PCS_NAD27_Texas_Central = 32039
PCS_NAD27_Texas_South_Cen =      32040
PCS_NAD27_Texas_South =   32041
PCS_NAD27_Utah_North =    32042
```

```
PCS_NAD27_Utah_Central =  32043
PCS_NAD27_Utah_South =    32044
PCS_NAD27_Vermont =       32045
PCS_NAD27_Virginia_North =       32046
PCS_NAD27_Virginia_South =       32047
PCS_NAD27_Washington_North =     32048
PCS_NAD27_Washington_South =     32049
PCS_NAD27_West_Virginia_N =      32050
PCS_NAD27_West_Virginia_S =      32051
PCS_NAD27_Wisconsin_North =      32052
PCS_NAD27_Wisconsin_Cen = 32053
PCS_NAD27_Wisconsin_South =      32054
PCS_NAD27_Wyoming_East =  32055
PCS_NAD27_Wyoming_E_Cen = 32056
PCS_NAD27_Wyoming_W_Cen = 32057
PCS_NAD27_Wyoming_West =  32058
PCS_NAD27_Puerto_Rico =   32059
PCS_NAD27_St_Croix =      32060
PCS_NAD83_Montana =       32100
PCS_NAD83_Nebraska =      32104
PCS_NAD83_Nevada_East =   32107
PCS_NAD83_Nevada_Central =       32108
PCS_NAD83_Nevada_West =   32109
PCS_NAD83_New_Hampshire = 32110
PCS_NAD83_New_Jersey =    32111
PCS_NAD83_New_Mexico_East =      32112
PCS_NAD83_New_Mexico_Cent =      32113
PCS_NAD83_New_Mexico_West =      32114
PCS_NAD83_New_York_East = 32115
PCS_NAD83_New_York_Central =     32116
PCS_NAD83_New_York_West = 32117
PCS_NAD83_New_York_Long_Is =     32118
PCS_NAD83_North_Carolina =       32119
PCS_NAD83_North_Dakota_N =       32120
PCS_NAD83_North_Dakota_S =       32121
PCS_NAD83_Ohio_North =    32122
PCS_NAD83_Ohio_South =    32123
PCS_NAD83_Oklahoma_North =       32124
PCS_NAD83_Oklahoma_South =       32125
PCS_NAD83_Oregon_North =  32126
PCS_NAD83_Oregon_South =  32127
PCS_NAD83_Pennsylvania_N =       32128
PCS_NAD83_Pennsylvania_S =       32129
PCS_NAD83_Rhode_Island =  32130
PCS_NAD83_South_Carolina =       32133
PCS_NAD83_South_Dakota_N =       32134
PCS_NAD83_South_Dakota_S =       32135
PCS_NAD83_Tennessee =     32136
PCS_NAD83_Texas_North =   32137
PCS_NAD83_Texas_North_Cen =      32138
PCS_NAD83_Texas_Central = 32139
PCS_NAD83_Texas_South_Cen =      32140
PCS_NAD83_Texas_South =   32141
PCS_NAD83_Utah_North =    32142
PCS_NAD83_Utah_Central =  32143
PCS_NAD83_Utah_South =    32144
PCS_NAD83_Vermont =       32145
```

```
PCS_NAD83_Virginia_North =       32146
PCS_NAD83_Virginia_South =       32147
PCS_NAD83_Washington_North =     32148
PCS_NAD83_Washington_South =     32149
PCS_NAD83_West_Virginia_N =      32150
PCS_NAD83_West_Virginia_S =      32151
PCS_NAD83_Wisconsin_North =      32152
PCS_NAD83_Wisconsin_Cen = 32153
PCS_NAD83_Wisconsin_South =      32154
PCS_NAD83_Wyoming_East =  32155
PCS_NAD83_Wyoming_E_Cen = 32156
PCS_NAD83_Wyoming_W_Cen = 32157
PCS_NAD83_Wyoming_West =  32158
PCS_NAD83_Puerto_Rico_Virgin_Is =       32161
PCS_WGS72_UTM_zone_1N =   32201
PCS_WGS72_UTM_zone_2N =   32202
PCS_WGS72_UTM_zone_3N =   32203
PCS_WGS72_UTM_zone_4N =   32204
PCS_WGS72_UTM_zone_5N =   32205
PCS_WGS72_UTM_zone_6N =   32206
PCS_WGS72_UTM_zone_7N =   32207
PCS_WGS72_UTM_zone_8N =   32208
PCS_WGS72_UTM_zone_9N =   32209
PCS_WGS72_UTM_zone_10N =  32210
PCS_WGS72_UTM_zone_11N =  32211
PCS_WGS72_UTM_zone_12N =  32212
PCS_WGS72_UTM_zone_13N =  32213
PCS_WGS72_UTM_zone_14N =  32214
PCS_WGS72_UTM_zone_15N =  32215
PCS_WGS72_UTM_zone_16N =  32216
PCS_WGS72_UTM_zone_17N =  32217
PCS_WGS72_UTM_zone_18N =  32218
PCS_WGS72_UTM_zone_19N =  32219
PCS_WGS72_UTM_zone_20N =  32220
PCS_WGS72_UTM_zone_21N =  32221
PCS_WGS72_UTM_zone_22N =  32222
PCS_WGS72_UTM_zone_23N =  32223
PCS_WGS72_UTM_zone_24N =  32224
PCS_WGS72_UTM_zone_25N =  32225
PCS_WGS72_UTM_zone_26N =  32226
PCS_WGS72_UTM_zone_27N =  32227
PCS_WGS72_UTM_zone_28N =  32228
PCS_WGS72_UTM_zone_29N =  32229
PCS_WGS72_UTM_zone_30N =  32230
PCS_WGS72_UTM_zone_31N =  32231
PCS_WGS72_UTM_zone_32N =  32232
PCS_WGS72_UTM_zone_33N =  32233
PCS_WGS72_UTM_zone_34N =  32234
PCS_WGS72_UTM_zone_35N =  32235
PCS_WGS72_UTM_zone_36N =  32236
PCS_WGS72_UTM_zone_37N =  32237
PCS_WGS72_UTM_zone_38N =  32238
PCS_WGS72_UTM_zone_39N =  32239
PCS_WGS72_UTM_zone_40N =  32240
PCS_WGS72_UTM_zone_41N =  32241
PCS_WGS72_UTM_zone_42N =  32242
PCS_WGS72_UTM_zone_43N =  32243
```

```
PCS_WGS72_UTM_zone_44N =  32244
PCS_WGS72_UTM_zone_45N =  32245
PCS_WGS72_UTM_zone_46N =  32246
PCS_WGS72_UTM_zone_47N =  32247
PCS_WGS72_UTM_zone_48N =  32248
PCS_WGS72_UTM_zone_49N =  32249
PCS_WGS72_UTM_zone_50N =  32250
PCS_WGS72_UTM_zone_51N =  32251
PCS_WGS72_UTM_zone_52N =  32252
PCS_WGS72_UTM_zone_53N =  32253
PCS_WGS72_UTM_zone_54N =  32254
PCS_WGS72_UTM_zone_55N =  32255
PCS_WGS72_UTM_zone_56N =  32256
PCS_WGS72_UTM_zone_57N =  32257
PCS_WGS72_UTM_zone_58N =  32258
PCS_WGS72_UTM_zone_59N =  32259
PCS_WGS72_UTM_zone_60N =  32260
PCS_WGS72_UTM_zone_1S =   32301
PCS_WGS72_UTM_zone_2S =   32302
PCS_WGS72_UTM_zone_3S =   32303
PCS_WGS72_UTM_zone_4S =   32304
PCS_WGS72_UTM_zone_5S =   32305
PCS_WGS72_UTM_zone_6S =   32306
PCS_WGS72_UTM_zone_7S =   32307
PCS_WGS72_UTM_zone_8S =   32308
PCS_WGS72_UTM_zone_9S =   32309
PCS_WGS72_UTM_zone_10S =  32310
PCS_WGS72_UTM_zone_11S =  32311
PCS_WGS72_UTM_zone_12S =  32312
PCS_WGS72_UTM_zone_13S =  32313
PCS_WGS72_UTM_zone_14S =  32314
PCS_WGS72_UTM_zone_15S =  32315
PCS_WGS72_UTM_zone_16S =  32316
PCS_WGS72_UTM_zone_17S =  32317
PCS_WGS72_UTM_zone_18S =  32318
PCS_WGS72_UTM_zone_19S =  32319
PCS_WGS72_UTM_zone_20S =  32320
PCS_WGS72_UTM_zone_21S =  32321
PCS_WGS72_UTM_zone_22S =  32322
PCS_WGS72_UTM_zone_23S =  32323
PCS_WGS72_UTM_zone_24S =  32324
PCS_WGS72_UTM_zone_25S =  32325
PCS_WGS72_UTM_zone_26S =  32326
PCS_WGS72_UTM_zone_27S =  32327
PCS_WGS72_UTM_zone_28S =  32328
PCS_WGS72_UTM_zone_29S =  32329
PCS_WGS72_UTM_zone_30S =  32330
PCS_WGS72_UTM_zone_31S =  32331
PCS_WGS72_UTM_zone_32S =  32332
PCS_WGS72_UTM_zone_33S =  32333
PCS_WGS72_UTM_zone_34S =  32334
PCS_WGS72_UTM_zone_35S =  32335
PCS_WGS72_UTM_zone_36S =  32336
PCS_WGS72_UTM_zone_37S =  32337
PCS_WGS72_UTM_zone_38S =  32338
PCS_WGS72_UTM_zone_39S =  32339
PCS_WGS72_UTM_zone_40S =  32340
```

```
PCS_WGS72_UTM_zone_41S =  32341
PCS_WGS72_UTM_zone_42S =  32342
PCS_WGS72_UTM_zone_43S =  32343
PCS_WGS72_UTM_zone_44S =  32344
PCS_WGS72_UTM_zone_45S =  32345
PCS_WGS72_UTM_zone_46S =  32346
PCS_WGS72_UTM_zone_47S =  32347
PCS_WGS72_UTM_zone_48S =  32348
PCS_WGS72_UTM_zone_49S =  32349
PCS_WGS72_UTM_zone_50S =  32350
PCS_WGS72_UTM_zone_51S =  32351
PCS_WGS72_UTM_zone_52S =  32352
PCS_WGS72_UTM_zone_53S =  32353
PCS_WGS72_UTM_zone_54S =  32354
PCS_WGS72_UTM_zone_55S =  32355
PCS_WGS72_UTM_zone_56S =  32356
PCS_WGS72_UTM_zone_57S =  32357
PCS_WGS72_UTM_zone_58S =  32358
PCS_WGS72_UTM_zone_59S =  32359
PCS_WGS72_UTM_zone_60S =  32360
PCS_WGS72BE_UTM_zone_1N = 32401
PCS_WGS72BE_UTM_zone_2N = 32402
PCS_WGS72BE_UTM_zone_3N = 32403
PCS_WGS72BE_UTM_zone_4N = 32404
PCS_WGS72BE_UTM_zone_5N = 32405
PCS_WGS72BE_UTM_zone_6N = 32406
PCS_WGS72BE_UTM_zone_7N = 32407
PCS_WGS72BE_UTM_zone_8N = 32408
PCS_WGS72BE_UTM_zone_9N = 32409
PCS_WGS72BE_UTM_zone_10N =       32410
PCS_WGS72BE_UTM_zone_11N =       32411
PCS_WGS72BE_UTM_zone_12N =       32412
PCS_WGS72BE_UTM_zone_13N =       32413
PCS_WGS72BE_UTM_zone_14N =       32414
PCS_WGS72BE_UTM_zone_15N =       32415
PCS_WGS72BE_UTM_zone_16N =       32416
PCS_WGS72BE_UTM_zone_17N =       32417
PCS_WGS72BE_UTM_zone_18N =       32418
PCS_WGS72BE_UTM_zone_19N =       32419
PCS_WGS72BE_UTM_zone_20N =       32420
PCS_WGS72BE_UTM_zone_21N =       32421
PCS_WGS72BE_UTM_zone_22N =       32422
PCS_WGS72BE_UTM_zone_23N =       32423
PCS_WGS72BE_UTM_zone_24N =       32424
PCS_WGS72BE_UTM_zone_25N =       32425
PCS_WGS72BE_UTM_zone_26N =       32426
PCS_WGS72BE_UTM_zone_27N =       32427
PCS_WGS72BE_UTM_zone_28N =       32428
PCS_WGS72BE_UTM_zone_29N =       32429
PCS_WGS72BE_UTM_zone_30N =       32430
PCS_WGS72BE_UTM_zone_31N =       32431
PCS_WGS72BE_UTM_zone_32N =       32432
PCS_WGS72BE_UTM_zone_33N =       32433
PCS_WGS72BE_UTM_zone_34N =       32434
PCS_WGS72BE_UTM_zone_35N =       32435
PCS_WGS72BE_UTM_zone_36N =       32436
PCS_WGS72BE_UTM_zone_37N =       32437
```

PCS_WGS72BE_UTM_zone_38N = 32438
PCS_WGS72BE_UTM_zone_39N = 32439
PCS_WGS72BE_UTM_zone_40N = 32440
PCS_WGS72BE_UTM_zone_41N = 32441
PCS_WGS72BE_UTM_zone_42N = 32442
PCS_WGS72BE_UTM_zone_43N = 32443
PCS_WGS72BE_UTM_zone_44N = 32444
PCS_WGS72BE_UTM_zone_45N = 32445
PCS_WGS72BE_UTM_zone_46N = 32446
PCS_WGS72BE_UTM_zone_47N = 32447
PCS_WGS72BE_UTM_zone_48N = 32448
PCS_WGS72BE_UTM_zone_49N = 32449
PCS_WGS72BE_UTM_zone_50N = 32450
PCS_WGS72BE_UTM_zone_51N = 32451
PCS_WGS72BE_UTM_zone_52N = 32452
PCS_WGS72BE_UTM_zone_53N = 32453
PCS_WGS72BE_UTM_zone_54N = 32454
PCS_WGS72BE_UTM_zone_55N = 32455
PCS_WGS72BE_UTM_zone_56N = 32456
PCS_WGS72BE_UTM_zone_57N = 32457
PCS_WGS72BE_UTM_zone_58N = 32458
PCS_WGS72BE_UTM_zone_59N = 32459
PCS_WGS72BE_UTM_zone_60N = 32460
PCS_WGS72BE_UTM_zone_1S = 32501
PCS_WGS72BE_UTM_zone_2S = 32502
PCS_WGS72BE_UTM_zone_3S = 32503
PCS_WGS72BE_UTM_zone_4S = 32504
PCS_WGS72BE_UTM_zone_5S = 32505
PCS_WGS72BE_UTM_zone_6S = 32506
PCS_WGS72BE_UTM_zone_7S = 32507
PCS_WGS72BE_UTM_zone_8S = 32508
PCS_WGS72BE_UTM_zone_9S = 32509
PCS_WGS72BE_UTM_zone_10S = 32510
PCS_WGS72BE_UTM_zone_11S = 32511
PCS_WGS72BE_UTM_zone_12S = 32512
PCS_WGS72BE_UTM_zone_13S = 32513
PCS_WGS72BE_UTM_zone_14S = 32514
PCS_WGS72BE_UTM_zone_15S = 32515
PCS_WGS72BE_UTM_zone_16S = 32516
PCS_WGS72BE_UTM_zone_17S = 32517
PCS_WGS72BE_UTM_zone_18S = 32518
PCS_WGS72BE_UTM_zone_19S = 32519
PCS_WGS72BE_UTM_zone_20S = 32520
PCS_WGS72BE_UTM_zone_21S = 32521
PCS_WGS72BE_UTM_zone_22S = 32522
PCS_WGS72BE_UTM_zone_23S = 32523
PCS_WGS72BE_UTM_zone_24S = 32524
PCS_WGS72BE_UTM_zone_25S = 32525
PCS_WGS72BE_UTM_zone_26S = 32526
PCS_WGS72BE_UTM_zone_27S = 32527
PCS_WGS72BE_UTM_zone_28S = 32528
PCS_WGS72BE_UTM_zone_29S = 32529
PCS_WGS72BE_UTM_zone_30S = 32530
PCS_WGS72BE_UTM_zone_31S = 32531
PCS_WGS72BE_UTM_zone_32S = 32532
PCS_WGS72BE_UTM_zone_33S = 32533
PCS_WGS72BE_UTM_zone_34S = 32534

```
PCS_WGS72BE_UTM_zone_35S =       32535
PCS_WGS72BE_UTM_zone_36S =       32536
PCS_WGS72BE_UTM_zone_37S =       32537
PCS_WGS72BE_UTM_zone_38S =       32538
PCS_WGS72BE_UTM_zone_39S =       32539
PCS_WGS72BE_UTM_zone_40S =       32540
PCS_WGS72BE_UTM_zone_41S =       32541
PCS_WGS72BE_UTM_zone_42S =       32542
PCS_WGS72BE_UTM_zone_43S =       32543
PCS_WGS72BE_UTM_zone_44S =       32544
PCS_WGS72BE_UTM_zone_45S =       32545
PCS_WGS72BE_UTM_zone_46S =       32546
PCS_WGS72BE_UTM_zone_47S =       32547
PCS_WGS72BE_UTM_zone_48S =       32548
PCS_WGS72BE_UTM_zone_49S =       32549
PCS_WGS72BE_UTM_zone_50S =       32550
PCS_WGS72BE_UTM_zone_51S =       32551
PCS_WGS72BE_UTM_zone_52S =       32552
PCS_WGS72BE_UTM_zone_53S =       32553
PCS_WGS72BE_UTM_zone_54S =       32554
PCS_WGS72BE_UTM_zone_55S =       32555
PCS_WGS72BE_UTM_zone_56S =       32556
PCS_WGS72BE_UTM_zone_57S =       32557
PCS_WGS72BE_UTM_zone_58S =       32558
PCS_WGS72BE_UTM_zone_59S =       32559
PCS_WGS72BE_UTM_zone_60S =       32560
PCS_WGS84_UTM_zone_1N =   32601
PCS_WGS84_UTM_zone_2N =   32602
PCS_WGS84_UTM_zone_3N =   32603
PCS_WGS84_UTM_zone_4N =   32604
PCS_WGS84_UTM_zone_5N =   32605
PCS_WGS84_UTM_zone_6N =   32606
PCS_WGS84_UTM_zone_7N =   32607
PCS_WGS84_UTM_zone_8N =   32608
PCS_WGS84_UTM_zone_9N =   32609
PCS_WGS84_UTM_zone_10N =  32610
PCS_WGS84_UTM_zone_11N =  32611
PCS_WGS84_UTM_zone_12N =  32612
PCS_WGS84_UTM_zone_13N =  32613
PCS_WGS84_UTM_zone_14N =  32614
PCS_WGS84_UTM_zone_15N =  32615
PCS_WGS84_UTM_zone_16N =  32616
PCS_WGS84_UTM_zone_17N =  32617
PCS_WGS84_UTM_zone_18N =  32618
PCS_WGS84_UTM_zone_19N =  32619
PCS_WGS84_UTM_zone_20N =  32620
PCS_WGS84_UTM_zone_21N =  32621
PCS_WGS84_UTM_zone_22N =  32622
PCS_WGS84_UTM_zone_23N =  32623
PCS_WGS84_UTM_zone_24N =  32624
PCS_WGS84_UTM_zone_25N =  32625
PCS_WGS84_UTM_zone_26N =  32626
PCS_WGS84_UTM_zone_27N =  32627
PCS_WGS84_UTM_zone_28N =  32628
PCS_WGS84_UTM_zone_29N =  32629
PCS_WGS84_UTM_zone_30N =  32630
PCS_WGS84_UTM_zone_31N =  32631
```

```
PCS_WGS84_UTM_zone_32N =  32632
PCS_WGS84_UTM_zone_33N =  32633
PCS_WGS84_UTM_zone_34N =  32634
PCS_WGS84_UTM_zone_35N =  32635
PCS_WGS84_UTM_zone_36N =  32636
PCS_WGS84_UTM_zone_37N =  32637
PCS_WGS84_UTM_zone_38N =  32638
PCS_WGS84_UTM_zone_39N =  32639
PCS_WGS84_UTM_zone_40N =  32640
PCS_WGS84_UTM_zone_41N =  32641
PCS_WGS84_UTM_zone_42N =  32642
PCS_WGS84_UTM_zone_43N =  32643
PCS_WGS84_UTM_zone_44N =  32644
PCS_WGS84_UTM_zone_45N =  32645
PCS_WGS84_UTM_zone_46N =  32646
PCS_WGS84_UTM_zone_47N =  32647
PCS_WGS84_UTM_zone_48N =  32648
PCS_WGS84_UTM_zone_49N =  32649
PCS_WGS84_UTM_zone_50N =  32650
PCS_WGS84_UTM_zone_51N =  32651
PCS_WGS84_UTM_zone_52N =  32652
PCS_WGS84_UTM_zone_53N =  32653
PCS_WGS84_UTM_zone_54N =  32654
PCS_WGS84_UTM_zone_55N =  32655
PCS_WGS84_UTM_zone_56N =  32656
PCS_WGS84_UTM_zone_57N =  32657
PCS_WGS84_UTM_zone_58N =  32658
PCS_WGS84_UTM_zone_59N =  32659
PCS_WGS84_UTM_zone_60N =  32660
PCS_WGS84_UTM_zone_1S =   32701
PCS_WGS84_UTM_zone_2S =   32702
PCS_WGS84_UTM_zone_3S =   32703
PCS_WGS84_UTM_zone_4S =   32704
PCS_WGS84_UTM_zone_5S =   32705
PCS_WGS84_UTM_zone_6S =   32706
PCS_WGS84_UTM_zone_7S =   32707
PCS_WGS84_UTM_zone_8S =   32708
PCS_WGS84_UTM_zone_9S =   32709
PCS_WGS84_UTM_zone_10S =  32710
PCS_WGS84_UTM_zone_11S =  32711
PCS_WGS84_UTM_zone_12S =  32712
PCS_WGS84_UTM_zone_13S =  32713
PCS_WGS84_UTM_zone_14S =  32714
PCS_WGS84_UTM_zone_15S =  32715
PCS_WGS84_UTM_zone_16S =  32716
PCS_WGS84_UTM_zone_17S =  32717
PCS_WGS84_UTM_zone_18S =  32718
PCS_WGS84_UTM_zone_19S =  32719
PCS_WGS84_UTM_zone_20S =  32720
PCS_WGS84_UTM_zone_21S =  32721
PCS_WGS84_UTM_zone_22S =  32722
PCS_WGS84_UTM_zone_23S =  32723
PCS_WGS84_UTM_zone_24S =  32724
PCS_WGS84_UTM_zone_25S =  32725
PCS_WGS84_UTM_zone_26S =  32726
PCS_WGS84_UTM_zone_27S =  32727
PCS_WGS84_UTM_zone_28S =  32728
```

```
PCS_WGS84_UTM_zone_29S =  32729
PCS_WGS84_UTM_zone_30S =  32730
PCS_WGS84_UTM_zone_31S =  32731
PCS_WGS84_UTM_zone_32S =  32732
PCS_WGS84_UTM_zone_33S =  32733
PCS_WGS84_UTM_zone_34S =  32734
PCS_WGS84_UTM_zone_35S =  32735
PCS_WGS84_UTM_zone_36S =  32736
PCS_WGS84_UTM_zone_37S =  32737
PCS_WGS84_UTM_zone_38S =  32738
PCS_WGS84_UTM_zone_39S =  32739
PCS_WGS84_UTM_zone_40S =  32740
PCS_WGS84_UTM_zone_41S =  32741
PCS_WGS84_UTM_zone_42S =  32742
PCS_WGS84_UTM_zone_43S =  32743
PCS_WGS84_UTM_zone_44S =  32744
PCS_WGS84_UTM_zone_45S =  32745
PCS_WGS84_UTM_zone_46S =  32746
PCS_WGS84_UTM_zone_47S =  32747
PCS_WGS84_UTM_zone_48S =  32748
PCS_WGS84_UTM_zone_49S =  32749
PCS_WGS84_UTM_zone_50S =  32750
PCS_WGS84_UTM_zone_51S =  32751
PCS_WGS84_UTM_zone_52S =  32752
PCS_WGS84_UTM_zone_53S =  32753
PCS_WGS84_UTM_zone_54S =  32754
PCS_WGS84_UTM_zone_55S =  32755
PCS_WGS84_UTM_zone_56S =  32756
PCS_WGS84_UTM_zone_57S =  32757
PCS_WGS84_UTM_zone_58S =  32758
PCS_WGS84_UTM_zone_59S =  32759
PCS_WGS84_UTM_zone_60S =  32760

+-----------------------------------+
```

6.3.3.2 Projection Codes

```
Note: Projections do not include GCS or PCS definitions. If
possible, use the PCS code for standard projected coordinate
systems, and use this code only if nonstandard datums are required.
```

```
Ranges:

   0 = undefined
   [    1,  9999] = Obsolete EPSG/POSC Projection codes
   [10000, 19999] = EPSG/POSC Projection codes
   32767          = user-defined
   [32768, 65535] = Private User Implementations

Special Ranges:

  US State Plane Format:    1sszz
          where ss is USC&GS State code
          zz is USC&GS zone code for NAD27 zones
          zz is (USC&GS zone code + 30) for NAD83 zones
```

```
  Larger zoned systems (16000-17999)
   UTM (North) Format:  160zz
   UTM (South) Format:  161zz
   zoned Universal Gauss-Kruger     Format:  162zz
   Universal Gauss-Kruger (unzoned) Format:  163zz
   Australian Map Grid       Format:  174zz
   Southern African STM      Format:  175zz

  Smaller zoned systems:     Format:  18ssz
            where ss is sequential system number
            z is zone code

  Single zone projections    Format:   199ss
            where ss is sequential system number

Values:

   Proj_Alabama_CS27_East =  10101
   Proj_Alabama_CS27_West =  10102
   Proj_Alabama_CS83_East =  10131
   Proj_Alabama_CS83_West =  10132
   Proj_Arizona_Coordinate_System_east =    10201
   Proj_Arizona_Coordinate_System_Central = 10202
   Proj_Arizona_Coordinate_System_west =    10203
   Proj_Arizona_CS83_east =  10231
   Proj_Arizona_CS83_Central =      10232
   Proj_Arizona_CS83_west =  10233
   Proj_Arkansas_CS27_North =       10301
   Proj_Arkansas_CS27_South =       10302
   Proj_Arkansas_CS83_North =       10331
   Proj_Arkansas_CS83_South =       10332
   Proj_California_CS27_I =  10401
   Proj_California_CS27_II = 10402
   Proj_California_CS27_III =       10403
   Proj_California_CS27_IV = 10404
   Proj_California_CS27_V =  10405
   Proj_California_CS27_VI = 10406
   Proj_California_CS27_VII =       10407
   Proj_California_CS83_1 =  10431
   Proj_California_CS83_2 =  10432
   Proj_California_CS83_3 =  10433
   Proj_California_CS83_4 =  10434
   Proj_California_CS83_5 =  10435
   Proj_California_CS83_6 =  10436
   Proj_Colorado_CS27_North =       10501
   Proj_Colorado_CS27_Central =     10502
   Proj_Colorado_CS27_South =       10503
   Proj_Colorado_CS83_North =       10531
   Proj_Colorado_CS83_Central =     10532
   Proj_Colorado_CS83_South =       10533
   Proj_Connecticut_CS27 =   10600
   Proj_Connecticut_CS83 =   10630
   Proj_Delaware_CS27 =      10700
   Proj_Delaware_CS83 =      10730
   Proj_Florida_CS27_East =  10901
   Proj_Florida_CS27_West =  10902
   Proj_Florida_CS27_North = 10903
```

```
Proj_Florida_CS83_East =  10931
Proj_Florida_CS83_West =  10932
Proj_Florida_CS83_North = 10933
Proj_Georgia_CS27_East =  11001
Proj_Georgia_CS27_West =  11002
Proj_Georgia_CS83_East =  11031
Proj_Georgia_CS83_West =  11032
Proj_Idaho_CS27_East =    11101
Proj_Idaho_CS27_Central = 11102
Proj_Idaho_CS27_West =    11103
Proj_Idaho_CS83_East =    11131
Proj_Idaho_CS83_Central = 11132
Proj_Idaho_CS83_West =    11133
Proj_Illinois_CS27_East = 11201
Proj_Illinois_CS27_West = 11202
Proj_Illinois_CS83_East = 11231
Proj_Illinois_CS83_West = 11232
Proj_Indiana_CS27_East =  11301
Proj_Indiana_CS27_West =  11302
Proj_Indiana_CS83_East =  11331
Proj_Indiana_CS83_West =  11332
Proj_Iowa_CS27_North =    11401
Proj_Iowa_CS27_South =    11402
Proj_Iowa_CS83_North =    11431
Proj_Iowa_CS83_South =    11432
Proj_Kansas_CS27_North =  11501
Proj_Kansas_CS27_South =  11502
Proj_Kansas_CS83_North =  11531
Proj_Kansas_CS83_South =  11532
Proj_Kentucky_CS27_North =       11601
Proj_Kentucky_CS27_South =       11602
Proj_Kentucky_CS83_North =       11631
Proj_Kentucky_CS83_South =       11632
Proj_Louisiana_CS27_North =      11701
Proj_Louisiana_CS27_South =      11702
Proj_Louisiana_CS83_North =      11731
Proj_Louisiana_CS83_South =      11732
Proj_Maine_CS27_East =    11801
Proj_Maine_CS27_West =    11802
Proj_Maine_CS83_East =    11831
Proj_Maine_CS83_West =    11832
Proj_Maryland_CS27 =      11900
Proj_Maryland_CS83 =      11930
Proj_Massachusetts_CS27_Mainland =      12001
Proj_Massachusetts_CS27_Island = 12002
Proj_Massachusetts_CS83_Mainland =      12031
Proj_Massachusetts_CS83_Island = 12032
Proj_Michigan_State_Plane_East = 12101
Proj_Michigan_State_Plane_Old_Central =  12102
Proj_Michigan_State_Plane_West = 12103
Proj_Michigan_CS27_North =       12111
Proj_Michigan_CS27_Central =     12112
Proj_Michigan_CS27_South =       12113
Proj_Michigan_CS83_North =       12141
Proj_Michigan_CS83_Central =     12142
Proj_Michigan_CS83_South =       12143
Proj_Minnesota_CS27_North =      12201
```

```
Proj_Minnesota_CS27_Central =    12202
Proj_Minnesota_CS27_South =      12203
Proj_Minnesota_CS83_North =      12231
Proj_Minnesota_CS83_Central =    12232
Proj_Minnesota_CS83_South =      12233
Proj_Mississippi_CS27_East =     12301
Proj_Mississippi_CS27_West =     12302
Proj_Mississippi_CS83_East =     12331
Proj_Mississippi_CS83_West =     12332
Proj_Missouri_CS27_East = 12401
Proj_Missouri_CS27_Central =     12402
Proj_Missouri_CS27_West = 12403
Proj_Missouri_CS83_East = 12431
Proj_Missouri_CS83_Central =     12432
Proj_Missouri_CS83_West = 12433
Proj_Montana_CS27_North = 12501
Proj_Montana_CS27_Central =      12502
Proj_Montana_CS27_South = 12503
Proj_Montana_CS83 =       12530
Proj_Nebraska_CS27_North =       12601
Proj_Nebraska_CS27_South =       12602
Proj_Nebraska_CS83 =      12630
Proj_Nevada_CS27_East =   12701
Proj_Nevada_CS27_Central =       12702
Proj_Nevada_CS27_West =   12703
Proj_Nevada_CS83_East =   12731
Proj_Nevada_CS83_Central =       12732
Proj_Nevada_CS83_West =   12733
Proj_New_Hampshire_CS27 = 12800
Proj_New_Hampshire_CS83 = 12830
Proj_New_Jersey_CS27 =    12900
Proj_New_Jersey_CS83 =    12930
Proj_New_Mexico_CS27_East =      13001
Proj_New_Mexico_CS27_Central =   13002
Proj_New_Mexico_CS27_West =      13003
Proj_New_Mexico_CS83_East =      13031
Proj_New_Mexico_CS83_Central =   13032
Proj_New_Mexico_CS83_West =      13033
Proj_New_York_CS27_East = 13101
Proj_New_York_CS27_Central =     13102
Proj_New_York_CS27_West = 13103
Proj_New_York_CS27_Long_Island = 13104
Proj_New_York_CS83_East = 13131
Proj_New_York_CS83_Central =     13132
Proj_New_York_CS83_West = 13133
Proj_New_York_CS83_Long_Island = 13134
Proj_North_Carolina_CS27 =       13200
Proj_North_Carolina_CS83 =       13230
Proj_North_Dakota_CS27_North =   13301
Proj_North_Dakota_CS27_South =   13302
Proj_North_Dakota_CS83_North =   13331
Proj_North_Dakota_CS83_South =   13332
Proj_Ohio_CS27_North =    13401
Proj_Ohio_CS27_South =    13402
Proj_Ohio_CS83_North =    13431
Proj_Ohio_CS83_South =    13432
Proj_Oklahoma_CS27_North =       13501
```

```
Proj_Oklahoma_CS27_South =       13502
Proj_Oklahoma_CS83_North =       13531
Proj_Oklahoma_CS83_South =       13532
Proj_Oregon_CS27_North =  13601
Proj_Oregon_CS27_South =  13602
Proj_Oregon_CS83_North =  13631
Proj_Oregon_CS83_South =  13632
Proj_Pennsylvania_CS27_North =   13701
Proj_Pennsylvania_CS27_South =   13702
Proj_Pennsylvania_CS83_North =   13731
Proj_Pennsylvania_CS83_South =   13732
Proj_Rhode_Island_CS27 =  13800
Proj_Rhode_Island_CS83 =  13830
Proj_South_Carolina_CS27_North = 13901
Proj_South_Carolina_CS27_South = 13902
Proj_South_Carolina_CS83 =       13930
Proj_South_Dakota_CS27_North =   14001
Proj_South_Dakota_CS27_South =   14002
Proj_South_Dakota_CS83_North =   14031
Proj_South_Dakota_CS83_South =   14032
Proj_Tennessee_CS27 =     14100
Proj_Tennessee_CS83 =     14130
Proj_Texas_CS27_North =   14201
Proj_Texas_CS27_North_Central =  14202
Proj_Texas_CS27_Central = 14203
Proj_Texas_CS27_South_Central =  14204
Proj_Texas_CS27_South =   14205
Proj_Texas_CS83_North =   14231
Proj_Texas_CS83_North_Central =  14232
Proj_Texas_CS83_Central = 14233
Proj_Texas_CS83_South_Central =  14234
Proj_Texas_CS83_South =   14235
Proj_Utah_CS27_North =    14301
Proj_Utah_CS27_Central =  14302
Proj_Utah_CS27_South =    14303
Proj_Utah_CS83_North =    14331
Proj_Utah_CS83_Central =  14332
Proj_Utah_CS83_South =    14333
Proj_Vermont_CS27 =       14400
Proj_Vermont_CS83 =       14430
Proj_Virginia_CS27_North =       14501
Proj_Virginia_CS27_South =       14502
Proj_Virginia_CS83_North =       14531
Proj_Virginia_CS83_South =       14532
Proj_Washington_CS27_North =     14601
Proj_Washington_CS27_South =     14602
Proj_Washington_CS83_North =     14631
Proj_Washington_CS83_South =     14632
Proj_West_Virginia_CS27_North =  14701
Proj_West_Virginia_CS27_South =  14702
Proj_West_Virginia_CS83_North =  14731
Proj_West_Virginia_CS83_South =  14732
Proj_Wisconsin_CS27_North =      14801
Proj_Wisconsin_CS27_Central =    14802
Proj_Wisconsin_CS27_South =      14803
Proj_Wisconsin_CS83_North =      14831
Proj_Wisconsin_CS83_Central =    14832
```

```
Proj_Wisconsin_CS83_South =      14833
Proj_Wyoming_CS27_East =  14901
Proj_Wyoming_CS27_East_Central = 14902
Proj_Wyoming_CS27_West_Central = 14903
Proj_Wyoming_CS27_West =  14904
Proj_Wyoming_CS83_East =  14931
Proj_Wyoming_CS83_East_Central = 14932
Proj_Wyoming_CS83_West_Central = 14933
Proj_Wyoming_CS83_West =  14934
Proj_Alaska_CS27_1 =      15001
Proj_Alaska_CS27_2 =      15002
Proj_Alaska_CS27_3 =      15003
Proj_Alaska_CS27_4 =      15004
Proj_Alaska_CS27_5 =      15005
Proj_Alaska_CS27_6 =      15006
Proj_Alaska_CS27_7 =      15007
Proj_Alaska_CS27_8 =      15008
Proj_Alaska_CS27_9 =      15009
Proj_Alaska_CS27_10 =     15010
Proj_Alaska_CS83_1 =      15031
Proj_Alaska_CS83_2 =      15032
Proj_Alaska_CS83_3 =      15033
Proj_Alaska_CS83_4 =      15034
Proj_Alaska_CS83_5 =      15035
Proj_Alaska_CS83_6 =      15036
Proj_Alaska_CS83_7 =      15037
Proj_Alaska_CS83_8 =      15038
Proj_Alaska_CS83_9 =      15039
Proj_Alaska_CS83_10 =     15040
Proj_Hawaii_CS27_1 =      15101
Proj_Hawaii_CS27_2 =      15102
Proj_Hawaii_CS27_3 =      15103
Proj_Hawaii_CS27_4 =      15104
Proj_Hawaii_CS27_5 =      15105
Proj_Hawaii_CS83_1 =      15131
Proj_Hawaii_CS83_2 =      15132
Proj_Hawaii_CS83_3 =      15133
Proj_Hawaii_CS83_4 =      15134
Proj_Hawaii_CS83_5 =      15135
Proj_Puerto_Rico_CS27 =   15201
Proj_St_Croix =    15202
Proj_Puerto_Rico_Virgin_Is =     15230
Proj_BLM_14N_feet =       15914
Proj_BLM_15N_feet =       15915
Proj_BLM_16N_feet =       15916
Proj_BLM_17N_feet =       15917
Proj_Map_Grid_of_Australia_48 =  17348
Proj_Map_Grid_of_Australia_49 =  17349
Proj_Map_Grid_of_Australia_50 =  17350
Proj_Map_Grid_of_Australia_51 =  17351
Proj_Map_Grid_of_Australia_52 =  17352
Proj_Map_Grid_of_Australia_53 =  17353
Proj_Map_Grid_of_Australia_54 =  17354
Proj_Map_Grid_of_Australia_55 =  17355
Proj_Map_Grid_of_Australia_56 =  17356
Proj_Map_Grid_of_Australia_57 =  17357
Proj_Map_Grid_of_Australia_58 =  17358
```

```
Proj_Australian_Map_Grid_48 =    17448
Proj_Australian_Map_Grid_49 =    17449
Proj_Australian_Map_Grid_50 =    17450
Proj_Australian_Map_Grid_51 =    17451
Proj_Australian_Map_Grid_52 =    17452
Proj_Australian_Map_Grid_53 =    17453
Proj_Australian_Map_Grid_54 =    17454
Proj_Australian_Map_Grid_55 =    17455
Proj_Australian_Map_Grid_56 =    17456
Proj_Australian_Map_Grid_57 =    17457
Proj_Australian_Map_Grid_58 =    17458
Proj_Argentina_1 = 18031
Proj_Argentina_2 = 18032
Proj_Argentina_3 = 18033
Proj_Argentina_4 = 18034
Proj_Argentina_5 = 18035
Proj_Argentina_6 = 18036
Proj_Argentina_7 = 18037
Proj_Colombia_3W = 18051
Proj_Colombia_Bogota =    18052
Proj_Colombia_3E = 18053
Proj_Colombia_6E = 18054
Proj_Egypt_Red_Belt =     18072
Proj_Egypt_Purple_Belt =  18073
Proj_Extended_Purple_Belt =      18074
Proj_New_Zealand_North_Island_Nat_Grid = 18141
Proj_New_Zealand_South_Island_Nat_Grid = 18142
Proj_Bahrain_Grid =       19900
Proj_Netherlands_E_Indies_Equatorial =   19905
Proj_RSO_Borneo =  19912


+-----------------------------------+
```

6.3.3.3 Coordinate Transformation Codes

```
Ranges:

   0 = undefined
   [    1, 16383] = GeoTIFF Coordinate Transformation codes
   [16384, 32766] = Reserved by GeoTIFF
   32767          = user-defined
   [32768, 65535] = Private User Implementations

Values:

   CT_TransverseMercator =   1
   CT_TransvMercator_Modified_Alaska = 2
   CT_ObliqueMercator =      3
   CT_ObliqueMercator_Laborde =     4
   CT_ObliqueMercator_Rosenmund =   5
   CT_ObliqueMercator_Spherical =   6
   CT_Mercator =      7
   CT_LambertConfConic_2SP = 8
   CT_LambertConfConic_Helmert =    9
```

```
   CT_LambertAzimEqualArea = 10
   CT_AlbersEqualArea =      11
   CT_AzimuthalEquidistant = 12
   CT_EquidistantConic =     13
   CT_Stereographic = 14
   CT_PolarStereographic =   15
   CT_ObliqueStereographic = 16
   CT_Equirectangular =      17
   CT_CassiniSoldner =       18
   CT_Gnomonic =      19
   CT_MillerCylindrical =    20
   CT_Orthographic =  21
   CT_Polyconic =     22
   CT_Robinson =      23
   CT_Sinusoidal =    24
   CT_VanDerGrinten = 25
   CT_NewZealandMapGrid =    26
   CT_TransvMercator_SouthOriented= 27
```

Aliases:

```
   CT_AlaskaConformal =              CT_TransvMercator_Modified_Alaska
   CT_TransvEquidistCylindrical =    CT_CassiniSoldner
   CT_ObliqueMercator_Hotine =       CT_ObliqueMercator
   CT_SwissObliqueCylindrical =      CT_ObliqueMercator_Rosenmund
   CT_GaussBoaga =                   CT_TransverseMercator
   CT_GaussKruger =                  CT_TransverseMercator
   CT_LambertConfConic =             CT_LambertConfConic_2SP
   CT_LambertConfConic_Helmert =     CT_LambertConfConic_1SP
   CT_SouthOrientedGaussConformal = CT_TransvMercator_SouthOriented
```

+-----------------------------------+

6.3.4 Vertical CS Codes

+-----------------------------------+

6.3.4.1 Vertical CS Type Codes

Ranges:

```
   0                = undefined
   [    1,   4999] = Reserved
   [ 5000,   5099] = EPSG Ellipsoid Vertical CS Codes
   [ 5100,   5199] = EPSG Orthometric Vertical CS Codes
   [ 5200,   5999] = Reserved EPSG
   [ 6000,  32766] = Reserved
   32767           = user-defined
   [32768, 65535]  = Private User Implementations
```

Values:

```
   VertCS_Airy_1830_ellipsoid =     5001
   VertCS_Airy_Modified_1849_ellipsoid =   5002
   VertCS_ANS_ellipsoid =    5003
   VertCS_Bessel_1841_ellipsoid =   5004
   VertCS_Bessel_Modified_ellipsoid =      5005
```

```
   VertCS_Bessel_Namibia_ellipsoid =       5006
   VertCS_Clarke_1858_ellipsoid =   5007
   VertCS_Clarke_1866_ellipsoid =   5008
   VertCS_Clarke_1880_Benoit_ellipsoid =   5010
   VertCS_Clarke_1880_IGN_ellipsoid =      5011
   VertCS_Clarke_1880_RGS_ellipsoid =      5012
   VertCS_Clarke_1880_Arc_ellipsoid =      5013
   VertCS_Clarke_1880_SGA_1922_ellipsoid =  5014
   VertCS_Everest_1830_1937_Adjustment_ellipsoid = 5015
   VertCS_Everest_1830_1967_Definition_ellipsoid = 5016
   VertCS_Everest_1830_1975_Definition_ellipsoid = 5017
   VertCS_Everest_1830_Modified_ellipsoid = 5018
   VertCS_GRS_1980_ellipsoid =      5019
   VertCS_Helmert_1906_ellipsoid =  5020
   VertCS_INS_ellipsoid =    5021
   VertCS_International_1924_ellipsoid =   5022
   VertCS_International_1967_ellipsoid =   5023
   VertCS_Krassowsky_1940_ellipsoid =      5024
   VertCS_NWL_9D_ellipsoid = 5025
   VertCS_NWL_10D_ellipsoid =       5026
   VertCS_Plessis_1817_ellipsoid =  5027
   VertCS_Struve_1860_ellipsoid =   5028
   VertCS_War_Office_ellipsoid =    5029
   VertCS_WGS_84_ellipsoid = 5030
   VertCS_GEM_10C_ellipsoid =       5031
   VertCS_OSU86F_ellipsoid = 5032
   VertCS_OSU91A_ellipsoid = 5033

  Orthometric Vertical CS;

   VertCS_Newlyn =    5101
   VertCS_North_American_Vertical_Datum_1929 =     5102
   VertCS_North_American_Vertical_Datum_1988 =     5103
   VertCS_Yellow_Sea_1956 =  5104
   VertCS_Baltic_Sea =       5105
   VertCS_Caspian_Sea =      5106


   +-----------------------------------+
```

6.3.4.2 Vertical CS Datum Codes

```
Ranges:

   0                = undefined
   [    1,  16383] = Vertical Datum Codes
   [16384,  32766] = Reserved
   32767           = user-defined
   [32768, 65535]  = Private User Implementations

No vertical datum codes are currently defined, other than those implied
by
the corrsponding Vertical CS code.

   +--------------------------------------------------------------------
   -+
```

+--------------------------------+

6.4 EPSG Geodesy Parameter Index

+--------------------------------+

Here is a summary of the index ranges for the various coding systems used by EPSG in their tables. A copy of this index may be acquired at the FTP sites mentioned in the references in section 5. The "value" table entries below describe how values from one table are related to codes from another table.

Summary

Entity	digit	Range
Prime Meridian	8	8000 thru 8999
Ellipsoid	7	7000 thru 7999
Geodetic Datum	6	6000 thru 6999
Vertical datum	5	5000 thru 5999
Geographic Coordinate System	4	4000 thru 4999
Projected Coordinate Systems	2 or 3	20000 thru 32760
Map Projection	1	10000 - 19999

Geodetic Datum Codes

Datum Type	Value	Range	Currently Defined
Unspecified Geodetic Datum	[EC-1000]	6000 thru 6099	6001 thru 6035
Geodetic Datum		6100 thru 6321	6200 thru 6315
WGS 72; WGS 72BE and WGS84		6322 thru 6327	6322 thru 6327
Geodetic Datum (ancient)		6900 thru 6999	6901 thru 6902

Note for Values: EC = corresponding Ellipsoid Code.

Vertical Datum Codes

Datum Type	Value	Range	Currently Defined
Ellipsoidal	[EC-1000]	5000 thru 5099	5001 thru 5035
Orthometric		5100 thru 5899	5101 thru 5106

Note for Values: EC = corresponding Ellipsoid Code.

Geographic Coordinate System Codes

GCS Type	Value	Range	Currently Defined
Unknown geodetic datum	[GDC-2000]	4000 thru 4099	4001 thru 4045
Known datum (Greenwich)	[GDC-2000]	4100 thru 4321	4200 thru 4315
WGS 72; WGS 72BE and WGS84		4322 thru 4327	4322 thru 4327
Known datum (not Greenwich)		4800 thru 4899	4801 thru 4812
Known datum (ancient)	[GDC-2000]	4900 thru 4999	4901 thru 4902

Note for Values: GDC = corresponding Geodetic Datum Code

Map Projection System Codes

```
   US State Plane  ( 10000-15999 )
      Format:     1sszz
         where ss is USC&GS State code  01 thru 59
         zz is (USC&GS zone code)      for NAD27 zones
         zz is (USC&GS zone code + 30) for NAD83 zones
```

Larger zoned systems (16000-17999)

System	Format	zz Range	
UTM (North)	160zz	01	60
UTM (South)	161zz	01	60
zoned Universal Gauss-Kruger	162zz	04	32
Universal Gauss-Kruger (unzoned)	163zz	04	3
Australian Map Grid	174zz	48	58
Southern African STM	175zz	13	35

```
   Smaller zoned systems  ( 18000-18999 )
       Format:  18ssz
           where ss is sequential system number  01   18
           z is zone code

   Single zone projections ( 19900-19999 )
        Format:   199ss
           where ss is sequential system number  00   25
```

Projected Coordinate Systems

For PCS utilising GeogCS with code in range 4201 through 4321 (i.e. geodetic datum code 6201 through 6319):

As far as is possible the PCS code will be of the format gggzz where ggg is (geodetic datum code -6000) and zz is zone.

For PCS utilising GeogCS with code out of range 4201 through 4321 (i.e.geodetic datum code 6201 through 6319):
PCS code 20xxx where xxx is a sequential number

```
WGS72 / UTM North     322zz where zz is UTM zone number   32201   32260
WGS72 / UTM South     323zz where zz is UTM zone number   32301   32360
WGS72BE / UTM North   324zz where zz is UTM zone number   32401   32460
WGS72BE / UTM South   325zz where zz is UTM zone number   32501   32560
WGS84 / UTM North     326zz where zz is UTM zone number   32601   32660
WGS84 / UTM South     327zz where zz is UTM zone number   32701   32760
US State Plane (NAD27)   267xx or 320xx where xx is a sequential number
US State Plane (NAD83)   269xx or 321xx where xx is a sequential number
```

```
+--------------------------------------------------------------------
---+
```

7 Glossary

```
+--------------------------------------------------------------------
-+
```

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

Cell: A rectangular area in Raster space, in which a single pixel value is filled.

Code: In GeoTIFF, a code is a value assigned to a GeoKey, and has one of 65536 possible values.

Coordinate System: A systematic way of assigning real (x,y,z..) coordinates to a surface or volume. In Geodetics the surface is an ellipsoid used to model the earth.

Datum: a mathematical approximation to all or part of the earth's surface. Defining a datum requires the definition of an ellipsoid, its location and orientation, as well as the area for which the datum is valid.

Device Space	A coordinate space referencing scanner, printers and display devices.
DOUBLE:	8-byte IEEE double precision floating point.
Ellipsoid:	A mathematically defined quadratic surface used to model the earth.
EPSG:	European Petroleum Survey Group.
Flattening:	For an ellipsoid with major and minor axis lengths (a,b), the flattening is defined by:, `f = (a - b)/a` For the earth, the value of f is approximately 1/298.3
Geocoding:	An image is geocoded if a precise algorithm for determining the earth-location of each point in the image is defined.
Geographic Coordinate System:	A Geographic CS consists 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.
GeoKey	In GeoTIFF, a GeoKey is equivalent in function to a TIFF tag, but uses a different storage mechanism.
Georeferencing:	An image is georeferenced if the location of its pixels in some model space is defined, but the transformation tying model space to the earth is not known.

GeoTIFF:	A standard for storing georeference and geocoding information in a TIFF 6.0 compliant raster file.
Grid	A coordinate mesh upon which pixels are placed
IEEE	Institute of Electrical and Electronics Engineers, Inc.
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).
Meridian:	Arc of constant longitude, passing through the poles.
Model Space	A flat geometrical space used to model a portion of the earth.
Parallel:	Lines of constant latitude, parallel to the equator.
Pixel:	A dimensionless point-measurement, stored in a raster file.
POSC:	Petrotechnical Open Software Corporation.
Prime Meridian:	An arbitrarily chosen meridian, used as reference for all others, and defined as 0 degrees longitude.
Projection	A projection in GeoTIFF consists of a linear (X,Y) coordinate system, and a coordinate transformation method (such as Transverse Mercator) to tie this system to an unspecified Geographic CS..

Projected Coordinate System	The result of the application of a projection transformation of a Geographic coordinate system
Raster Space:	A continuous planar space in which pixel values are visually realized.
RATIONAL:	In TIFF format, a RATIONAL value is a fractional value represented by the ratio of two unsigned 4-byte integers.
SDTS	The USGS Spatial Data Transmission Standard.
Tag:	In TIFF format, a tag is packet of numerical or ASCII values, which have a numerical "Tag" ID indicating their information content.
TIFF:	Acronym for Tagged Image File Format; a platform-independent, extensive specification for storing raster data and ancillary information in a single file.
USGS	US Geological Survey

+---
--+,

END OF SPECIFICATION

+---
--+