* Tag Specifications

# Chapter Overview

This chapter and the next address issues related to HDF tags and the data they represent. The first section of this chapter provides general information about tags and their interpretation. The remainder of the chapter contains a complete list of the HDF basic tags supported by HDF Version 4.1r3 and detailed tag specifications. The next chapter, Extended Tags and Special Elements, provides detailed information regarding HDF-supported extended tags and the special elements they define.

## The HDF Tag Space

As discussed in Chapter 2, "Basic Structure of HDF Files," 16 bits are allotted for an HDF tag number. This provides for 65535 possible tags, ranging from 1 to 65535; zero (0) is not used. This tag space is divided into three ranges:

1 – 32767 Reserved for HDF-supported tags

32768 – 64999 Set aside as user-definable tags

65000 – 65535 Reserved for expansion of the format

No restrictions are placed on the user-definable tags. Note that tags from this range are not expected to be unique across user-developed HDF applications.

The rest of this chapter is devoted to the HDF-supported basic tags in the range 1 (0x0001) to 16383 (0x3FFF). The next chapter, Extended Tags and Special Elements, is devoted to HDF-supported extended tags in the range 16384 (0x4000) to 32767 (0x7FFF).

## Tag Specifications

The following pages contain the specifications of the HDF-supported basic tags in HDF Version 4.1r3. Each entry contains the following information:

* The tag (in capital letters in the left margin)
* The full name of the tag (on the first line to the right)
* The type and, where possible, the amount of data in the corresponding data element (on the second line to the right)  
    
  When the data element is a variable-sized data structure—such as text, a string, or a variable-sized array—the amount of data cannot be specified exactly. Where possible, a formula is provided to estimate the amount of data. The string “? Bytes” appears when neither the size nor the structure of the data element can be specified.
* The tag number in decimal/(hexadecimal) (on the third line to the right)
* A diagram illustrating the structure of the tag and its associated data  
    
  Since all DDs that point to a data element contain data length and data offset fields, these fields are not included in the illustrations.
* A full specification of the tag, including a description of the data element and a discussion of its intended use.

Tags are roughly grouped according to the roles they play:

* Utility tags
* Annotation tags
* Compression tags
* Raster Image tags
* Composite image tags
* Vector image tags
* Scientific data set tags
* Vset tags
* Obsolete tags
* Extended tags (see Chapter 10, "Extended Tags and Special Elements)

These groupings imply a general context for the use of each tag; they are not meant to restrict their use.

Please note Section 9.3.9, "Obsolete Tags." These tags have fallen out of use with the continuing development of HDF. They are still recognized by the HDF library, but users should not write new objects using them; they may eventually be dropped from the HDF specification.

In the following discussion, the ground symbol indicates that the DD for this tag includes no pointer to a data element. I.e., there is never a data element associated with the tag.



### Utility Tags

DFTAG\_NULL No data  
0 bytes  
1 (0x0001)



*ref\_no* Reference number (16-bit integer; always 0)

This tag is used for place holding and to fill empty portions of the data description block. The length and offset fields (not shown) of a DFTAG\_NULL DD must be zero (0).

DFTAG\_VERSION Library version number  
12 bytes plus the length of a string  
30 (0x001E)



*ref\_no* Reference number (16-bit integer)

*majorv* Major version number (32-bit integer)

*minorv* Minor version number (32-bit integer)

*release* Release number (32-bit integer)

*string* Non-null terminated ASCII string (any length)

The data portion of this tag contains the complete version number and a descriptive string for the latest version of the HDF library to write to the file.

DFTAG\_NT Number type  
4 bytes  
106 (0x006A)



*ref\_no* Reference number (16-bit integer)

*version* Version number of NT information (8-bit integer)

*type* Unsigned integer, signed integer, unsigned character, character, floating point, double precision floating point (8-bit code)

*width* Number of bits, all of which are assumed to be significant (8-bit code)

*class* A generic value, with different interpretations depending on type: floating point, integer, or character (8-bit code)

Several values that may be used for each of the three types in the field CLASS are listed in Table 9a. This is not an exhaustive list.

* Number Type Values

|  |  |  |
| --- | --- | --- |
| Type | Mnemonic | Value |
| Floating point | DFNTF\_NONE | 0 |
| DFNTF\_IEEE | 1 |
| DFNTF\_VAX | 2 |
| DFNTF\_CRAY | 3 |
| DFNTF\_PC | 4 |
| DFNTF\_CONVEX | 5 |
| Integer | DFNTI\_MBO | 1 |
| DFNTI\_IBO | 2 |
| DFNTI\_VBO | 4 |
| Character | DFNTC\_ASCII1 | 1 |
| DFNTC\_EBCDOC | 2 |
| DFNTC\_BYTE | 0 |

The number type flag is used by any other element in the file to indicate specifically what a numeric value looks like. Other tag types should contain a reference number pointer to an DFTAG\_NT instead of containing their own number type definitions.

The version field allows expansion of the number type information, in case some future number types cannot be described using the fields currently defined. Successive versions of the DFTAG\_NT may be substantially different from the current definition, but backward compatibility will be maintained. The current DFTAG\_NT version number is 1.

DFTAG\_MT Machine type  
0 bytes  
107 (0x006B)



*double* Specifies method of encoding double precision floating point (4-bit code)

*float* Specifies method of encoding single precision floating point (4-bit code)

*int* Specifies method of encoding integers (4-bit code)

*char* Specifies method of encoding characters (4-bit code)

DFTAG\_MT specifies that all unconstrained or partially constrained values in this HDF file are of the default type for that hardware. When DFTAG\_MT is set to VAX, for example, all integers will be assumed to be in VAX byte order unless specifically defined otherwise with a DFTAG\_NT tag. Note that all of the headers and many tags, the whole raster image set for example, are defined with bit-wise precision and will not be overridden by the DFTAG\_MT setting.

For DFTAG\_MT, the reference field itself is the encoding of the DFTAG\_MT information. The reference field is 16 bits, taken as four groups of four bits, specifying the types for double-precision floating point, floating point, integer, and character respectively. This allows 16 generic specifications for each type.

To the user, these will be defined constants in the header file hdf.h, specifying the proper descriptive numbers for Sun, VAX, Cray, Convex, and other computer systems. If there is no DFTAG\_MT in a file, the application may assume that the data in the file has been written on the local machine; any portability problems must be addressed by the user. For this reason, we recommend that all HDF files contain a DFTAG\_MT for maximum portability.

Currently available data encodings are listed in Table 9L.

* Available Machine Types

|  |  |
| --- | --- |
| Type | Available Encodings |
| Double precision floating point | IEEE64  VAX64  CRAY128 |
| Floating point | IEEE32  VAX32  CRAY64 |
| Integers | VAX32  Intel16  Intel32  Motorola32  CRAY64 |
| Characters | ASCII  EBCDIC |

New encodings can be added for each data type as the need arises.

### Annotation Tags

DFTAG\_FID File identifier  
String  
100 (0x0064)



*ref\_no* Reference number (16-bit integer)

*character\_string*Non-null terminated ASCII text (any length)

This tag points to a string which the user wants to associate with this file. The string is not null terminated. The string is intended to be a user-supplied title for the file.

DFTAG\_FD File description  
Text  
101 (0x0065)



*ref\_no* Reference number (16-bit integer)

*text\_block* Non-null terminated ASCII text (any length)

This tag points to a block of text describing the overall file contents. The text can be any length. The block is not null terminated. The text is intended to be user-supplied comments about the file.

DFTAG\_TID Tag identifier   
String  
102 (0x0066)



*tag* Tag number to which this tag refers (16-bit integer)

*character\_string*Non-null terminated ASCII text (any length)

The data for this tag is a string that identifies the functionality of the tag indicated in the space normally used for the reference number. For example, the tag identifier for DFTAG\_TID might point to data that reads "tag identifier."

Many tags are identified in the HDF specification, so it is usually unnecessary to include their identifiers in the HDF file. But with user-defined tags or special-purpose tags, the only way for a human reader to diagnose what kind of data is stored in a file is to read tag identifiers. Use tag descriptions to define even more detail about your user-defined tags.

Note that with this tag you may make use of the user-defined tags to check for consistency. Although two persons may use the same user-defined tag, they probably will not use the same tag identifier.

DFTAG\_TD Tag description  
Text  
103 (0x0067)



*tag* Tag number to which this tag refers (16-bit integer)

*text\_block* Non-null terminated ASCII text (any length)

The data for this tag is a text block which describes in relative detail the functionality and format of the tag which is indicated in the space normally occupied by the reference number. This tag is intended to be used with user-defined tags and provides a medium for users to exchange files that include human-readable descriptions of the data.

It is important to provide everything that a programmer might need to know to read the data from your user-defined tag. At the minimum, you should specify everything you would need to know in order to retrieve your data at a later date if the original program were lost.

DFTAG\_DIL Data identifier label  
String  
104 (0x0068)



*ref\_no* Reference number (16-bit integer)

*obj\_tag* Tag number of the data to which this label applies (16-bit integer)

*obj\_ref\_no* Reference number of the data object to which this label applies (16-bit integer)

*character\_string*Non-null terminated ASCII text (any length)

The DFTAG\_DIL data object consists of a tag/ref followed by a string. The string serves as a label for the data identified by the tag/ref.

By including DFTAG\_DILtags, you can give a data object a label for future reference. For example, DFTAG\_DIL can be used to assign titles to images.

DFTAG\_DIA Data identifier annotation  
Text  
105 (0x0069)



*ref\_no* Reference number (16-bit integer)

*obj\_tag* Tag number of the data to which this annotation applies (16-bit integer)

*obj\_ref\_no* Reference number of the data object to which this annotation applies (16-bit integer)

*text\_block* Non-null terminated ASCII text (any length)

The DFTAG\_DIA data object consists of a tag/ref followed by a text block. The text block serves as an annotation of the data identified by the tag/ref.

With a DFTAG\_DIA tag, any data object can have a lengthy, user-written description. This can be used to include comments about images, data sets, source code, and so forth.

### Compression Tags

DFTAG\_RLE Run length encoded data  
0 bytes  
11 (0x000B)



*ref\_no* Reference number (16-bit integer)

This tag is used in the DFTAG\_ID compression field and in other places to indicate that an image or section of data is encoded with a run-length encoding scheme. The RLE method used is byte-wise. Each run is preceded by a count byte. The low seven bits of the count byte indicate the number of bytes (n). The high bit of the count byte indicates whether the next byte should be replicated *n* times (high bit = 1), or whether the next *n* bytes should be included as is (high bit = 0).

See also: DFTAG\_ID in “Raster Image Tags”

DFTAG\_NDG in “Scientific Data Set Tags”

DFTAG\_IMC IMCOMP compressed data  
0 bytes  
12 (0x000C)



*ref\_no* Reference number (16-bit integer)

This tag is used in the DFTAG\_ID compression field and in other places to indicate that an image or section of data is encoded with an IMCOMP encoding scheme. This scheme is a 4:1 aerial averaging method which is easy to decompress. It counts color frequencies in 4x4 squares to optimize color sampling.

See also: DFTAG\_ID in “Raster Image Tags”

DFTAG\_NDG in “Scientific Data Set Tags”

DFTAG\_JPEG 24-bit JPEG compression information  
? bytes  
13 (0x000D)



*ref\_no* Reference number (16-bit integer)

This tag is a flag indicating that the corresponding compressed object is a 24-bit JPEG image. The DFTAG\_JPEG flag and the corresponding DFTAG\_CI object will share the same reference number.

DFTAG\_GREYJPEG 8-bit JPEG compression information  
? bytes  
14 (0x000E)



*ref\_no* Reference number (16-bit integer)

This tag is a flag indicating that the corresponding compressed object is an 8-bit JPEG image. The DFTAG\_GREYJPEG flag and the corresponding DFTAG\_CI object will share the same reference number.

DFTAG\_CI Compressed raster image  
? bytes   
303 (0x012F



*ref\_no* Reference number (16-bit integer)

This tag points to a stream of bytes that make up a compressed image. The type of compression, together with any necessary parameters, are stored as a separate data object. For example, if DFTAG\_JPEG is contained in the same raster image group, the stream of bytes contains the JFIF header and all further data for the JPEG image. Other parameters are stored in the DFTAG\_JPEG object.

The JFIF header is the header data stored in a JFIF (JPEG File Interchange Format) file up to the start-of-frame parameter. See the document *JPEG File Interchange Format*[[1]](#footnote-1) for a detailed description of the file format.

### Raster Image Tags

DFTAG\_RIG Raster image group  
*n*\*4 bytes (where *n* is the number of data objects in the group)  
306 (0x0132)



*ref\_no* Reference number (16-bit integer)

*tag\_n* Tag number for nth member of the group (16-bit integer)

*ref\_n* Reference number for nth member of the group (16-bit integer)

The RIG data element contains the tag/refs of all the data objects required to display a raster image correctly. Application programs that deal with RIGs should read all the elements of a RIG and process those identifiers which it can display correctly. Even if the application cannot process *all* of the objects, the objects that it can process will be usable.

Table 9M lists the tags that may appear in an RIG.

* Available RIG Tags

|  |  |
| --- | --- |
| Tag | Description |
| DFTAG\_ID | Image dimension record |
| DFTAG\_RI | Raster image |
| DFTAG\_XYP | X-Y position |
| DFTAG\_LD | LUT dimension |
| DFTAG\_LUT | Color lookup table |
| DFTAG\_MD | Matte channel dimension |
| DFTAG\_MA | Matte channel |
| DFTAG\_CCN | Color correction |
| DFTAG\_CFM | Color format |
| DFTAG\_AR | Aspect ratio |

Example

DFTAG\_ID, DFTAG\_RI, DFTAG\_LD, DFTAG\_LUT

Assume that an image dimension record, a raster image, an LUT dimension record, and an LUT are all required to display a particular raster image correctly. These data objects can be associated in an RIG so that an application can read the image dimensions then the image. It will then read the lookup table and display the image.

DFTAG\_ID Image dimension  
20 bytes  
300 (0x012C)

DFTAG\_LD LUT dimension  
20 bytes  
307 (0x0133)

DFTAG\_MD Matte dimension  
20 bytes  
308 (0x0134)



*ref\_no* Reference number (16-bit integer)

*x\_dim* Length of x (horizontal) dimension (32-bit integer)

*y\_dim* Length of y (vertical) dimension (32-bit integer)

*NT\_ref* Reference number for number type information

*elements* Number of elements that make up one entry (16-bit integer)

*interlace* Type of interlacing used (16-bit integer)

0 The components of each pixel are together.

1 Color elements are grouped by scan lines.

2 Color elements are grouped by planes.

*comp\_tag* Tag which tells the type of compression used and any associated parameters (16-bit integer)

*comp\_ref* Reference number of compression tag (16-bit integer)

These three dimension records have exactly the same format; they specify the dimensions of the 2-dimensional arrays after which they are named and provide information regarding other attributes of the data in the array:

* DFTAG\_ID specifies the dimensions of a DFTAG\_RI.
* DFTAG\_LD specifies the dimensions of a DFTAG\_LUT.
* DFTAG\_MD specifies the dimensions of a DFTAG\_MA.

Other attributes described in the image dimension record include the number type of the elements, the number of elements per pixel, the interlace scheme used, and the compression scheme used (if any).

For example, a 512x256 row-wise 24-bit raster image with each pixel stored as RGB bytes would have the following values:

*x\_dim* 512

*y\_dim* 256

*NT\_ref* UINT8

*elements* 3 (3 elements per pixel: e.g., R, G, and B)

*interlace* 0 (RGB values not separated)

*comp\_tag* 0 (no compression is used)

The diagram above illustrates the tag DFTAG\_ID. The DFTAG\_LD and DFTAG\_MD diagrams would be identical except for the tag name in the first cell, which would be DFTAG\_LD and DFTAG\_MD, respectively.

DFTAG\_RI Raster image  
*xdim*\**ydim*\**elements*\**NTsize* bytes (*xdim*, *ydim*, *elements,*   
 and *NTsize* are specified in the corresponding DFTAG\_ID)  
302 (0x012E)



*ref\_no* Reference number (16-bit integer)

This tag points to raster image data. It is stored in row-major order and must be interpreted as specified by *interlace* in the related DFTAG\_ID.

DFTAG\_LUT Lookup table  
*xdim*\**ydim*\**elements*\**NTsize* bytes (*xdim*, *ydim*, *elements,*   
 and *NTsize* are specified in the corresponding DFTAG\_LD)  
301 (0x012D)



*ref\_no* Reference number (16-bit integer)

*Pn\_m* mth value of parameter n (size is specified by the DFTAG\_NT in the corresponding DFTAG\_LD)

The DFTAG\_LUT, sometimes called a palette, is used to assign colors to data values. When a raster image consists of data values which are going to be interpreted through an LUT capability, the DFTAG\_LUT should be loaded along with the image.

The most common lookup table is the RGB lookup table which will have X dimension = 256 and Y dimension = 1 with three elements per entry, one each for red, green, and blue. The interlace will be either 0, where the LUT values are given RGB, RGB, RGB, ..., or 1, where the LUT values are given as 256 reds, 256 greens, 256 blues.

DFTAG\_MA Matte channel  
*xdim*\**ydim*\**elements*\**NTsize* bytes (*xdim*, *ydim*, *elements,*   
 and *NTsize* are specified in the corresponding DFTAG\_MD)  
309 (0x0135)



*ref\_no* Reference number (16-bit integer)

The DFTAG\_MA data object contains transparency data which can be used to facilitate the overlaying of images. The data consists of a 2-dimensional array of unsigned 8-bit integers ranging from 0 to 255. Each point in a DFTAG\_MA indicates the transparency of the corresponding point in a raster image of the same dimensions. A value of 0 indicates that the data at that point is to be considered totally transparent, while a value of 255 indicates that the data at that point is totally opaque. It is assumed that a linear scale applies to the transparency values, but users may opt to interpret the data in any way they wish.

DFTAG\_CCN Color correction  
52 bytes (usually)  
310 (0x0136)



*ref\_no* Reference number (16-bit integer)

*gamma* Gamma parameter (32-bit IEEE floating point)

*red\_x*, *red\_y*, and *red\_z*Red x, y, and z correction factors (32-bit IEEE floating point)

*green\_x*, *green\_y*, and *green\_z*Green x, y, and z correction factors (32-bit IEEE floating point)

*blue\_x*, *blue\_y*, and *blue\_z*Blue x, y, and z correction factors (32-bit IEEE floating point)

*white\_x*, *white\_y*, and *white\_z*White x, y, and z correction factors (32-bit IEEE floating point)

Color correction specifies the Gamma correction for the image and color primaries for the generation of the image.

DFTAG\_CFM Color format  
String  
311 (0x0137)



*ref\_no* Reference number (16-bit integer)

*character\_string* Non-null terminated ASCII string (any length)

The color format data element contains a string of uppercase characters that indicates how each element of each pixel in a raster image is to be interpreted. Table 9N lists the available color format strings.

* Color Format String Values

|  |  |
| --- | --- |
| String | Description |
| VALUE | Pseudo-color, or just a value associated with the pixel |
| RGB | Red, green, blue model |
| XYZ | Color-space model |
| HSV | Hue, saturation, value model |
| HSI | Hue, saturation, intensity |
| SPECTRAL | Spectral sampling method |

DFTAG\_AR Aspect ratio  
4 bytes  
312 (0x0138)



*ref\_no* Reference number (16-bit integer)

*ratio* Ratio of width to height (32-bit IEEE float)

The data for this tag is the visual aspect ratio for this image. The image should be visually correct if displayed on a screen with this aspect ratio. The data consists of one floating-point number which represents width divided by height. An aspect ratio of 1.0 indicates a display with perfectly square pixels; 1.33 is a standard aspect ratio used by many monitors.

### Composite Image Tags

DFTAG\_DRAW Draw  
*n*\*4 bytes (where *n* is the number of data objects that make up   
 the composite image)  
400 (0x0190)



*ref\_no* Reference number (16-bit integer)

*tag\_n* Tag number of the nth member of the draw list (16-bit integer)

*ref\_n* Reference number of the nth member of the draw list (16-bit integer)

The DFTAG\_DRAW data element consists of a list of tag/refs that define a composite image. The data objects indicated should be displayed in order. This can include several RIGs which are to be displayed simultaneously. It can also include vector overlays, like DFTAG\_T14, which are to be placed on top of an RIG.

Some of the elements in a DFTAG\_DRAW list may be instructions about how images are to be composited (XOR, source put, anti-aliasing, etc.). These are defined as individual tags.

DFTAG\_XYP XY position  
8 bytes  
500 (0x01F4)



*ref\_no* Reference number (16-bit integer)

*x* X-coordinate (32-bit integer)

*y* Y-coordinate (32-bit integer)

DFTAG\_XYP is used in composites and other groups to indicate an XY position on the screen. For this, (0,0) is the lower left corner of the print area. X is the number of pixels to the right along the horizontal axis and Y is the number of pixels up on the vertical axis. The X and Y coordinates are two 32-bit integers.

For example, if DFTAG\_XYP is present in a DFTAG\_RIG, the DFTAG\_XYP specifies the position of the lower left corner of the raster image on the screen.

See also: DFTAG\_DRAW in this section

### Vector Image Tags

DFTAG\_T14 Tektronix 4014  
? bytes  
602 (0x25A)



*ref\_no* Reference number (16-bit integer)

This tag points to a Tektronix 4014 data stream. The bytes in the data field, when read and sent to a Tektronix 4014 terminal, will display a vector image. Only the lower seven bits of each byte are significant. There are no record markings or non-Tektronix codes in the data.

DFTAG\_T105 Tektronix 4105  
? bytes  
603 (0x25B)



*ref\_no* Reference number (16-bit integer)

This tag points to a Tektronix 4105 data stream. The bytes in the data field, when read and sent to a Tektronix 4105 terminal, will be displayed as a vector image. Only the lower seven bits of each byte are significant. Some terminal emulators will not correctly interpret every feature of the Tektronix 4105 terminal, so you may wish to use only a subset of the available Tektronix 4105 vector commands.

### Scientific Data Set Tags

DFTAG\_NDG Numeric data group  
*n*\*4 bytes (where *n* is the number of data objects in the group.)  
720 (0x02D0)



*ref\_no* Reference number (16-bit integer)

*tag\_n* Tag number of nth member of the group (16-bit integer)

*ref\_n* Reference number of nth member of the group   
(16-bit integer)

The NDG data contains a list of tag/refs that define a scientific data set. DFTAG\_NDG supersedes the old DFTAG\_SDG, which became obsolete upon the release on HDF Version 3.2. A more complete explanation of the relationship between DFTAG\_NDG and DFTAG\_SDG can be found in Chapter , “Sets and Groups.”

All of the members of an NDG provide information for correctly interpreting and displaying the data. Application programs that deal with NDGs should read all of the elements of a NDG and process those data objects which it can use. Even if an application cannot process all of the objects, the objects that it can understand will be usable.

Table 9O lists the tags that may appear in an NDG.

* Available NDG Tags

|  |  |
| --- | --- |
| Tag | Description |
| DFTAG\_SDD | Scientific data dimension record (rank and dimensions) |
| DFTAG\_SD | Scientific data |
| DFTAG\_SDS | Scales |
| DFTAG\_SDL | Labels |
| DFTAG\_SDU | Units |
| DFTAG\_SDF | Formats |
| DFTAG\_SDM | Maximum and minimum values |
| DFTAG\_SDC | Coordinate system |
| DFTAG\_CAL | Calibration information |
| DFTAG\_FV | Fill value |
| DFTAG\_LUT | Color lookup table |
| DFTAG\_LD | Lookup table dimension record |
| DFTAG\_SDLNK | Link to old-style DFTAG\_SDG |

Example

DFTAG\_SDD, DFTAG\_SD, DFTAG\_SDM

Suppose that an NDG contains a dimension record, scientific data, and the maximum and minimum values of the data. These data objects can be associated in an NDG so that an application can read the rank and dimensions from the dimension record and then read the data array. If the application needs maximum and minimum values, it will read them as well.

See also: Chapter , "Sets and Groups”

DFTAG\_SDD Scientific data dimension record  
6 + 8\**rank* bytes   
701 (0x02BD)



*ref\_no* Reference number (16-bit integer)

*rank* Number of dimensions (16-bit integer)

*dim\_n* Number of values along the nth dimension (32-bit integer)

*data\_NT\_ref* Reference number of DFTAG\_NT for data   
(16-bit integer)

*scale\_NT\_ref\_n*   
Reference number for DFTAG\_NT for the scale for the nth dimension (16-bit integer)

This record defines the rank and dimensions of the array in the scientific data set. For example, a DFTAG\_SDD for a 500x600x3 array of floating-point numbers would have the following values and components.

* Rank: 3
* Dimensions: 500, 600, and 3.
* One data NT
* Three scale NTs

DFTAG\_SD Scientific data  
*NTsize*\**x*\**y*\**z*\*... bytes (where NTsize is the size of the   
 data NT specified in the corresponding DFTAG\_SDD and   
 *x, y, z,* etc. are the dimension sizes)  
702 (0x02BE)



*ref\_no* Reference number (16-bit integer)

This tag points to an array of scientific data. The type of the data may be specified by an DFTAG\_NT included with the SDG. If there is no DFTAG\_NT, the type of the data is floating-point in standard IEEE 32-bit format. The rank and dimensions must be stored as specified in the corresponding DFTAG\_SDD. The diagram above shows a 3-dimensional data array.

DFTAG\_SDS Scientific data scales  
*rank* + *NTsize0*\**x* + *NTsize1*\**y* +*NTsize2*\**z* +... bytes (where *rank*   
 is the number of dimensions, *x*, *y*, *z*, etc. are the dimension  
 sizes, and *NTsize#* are the sizes of each scale NT from the   
 corresponding DFTAG\_SDD)  
703 (0x02BF)



*ref\_no* Reference number (16-bit integer)

*is\_n* A flag indicating whether a scale exists for the nth dimension (8-bit integer; 0 or 1)

*scale\_n* List of scale values for the nth dimension (type specified in corresponding DFTAG\_SDD)

This tag points to the scales for the data set. The first *n* bytes indicate whether there is a scale for the corresponding dimension (1 = yes, 0 = no). This is followed by the scale values for each dimension. The scale consists of a simple series of values where the number of values and their types are specified in the corresponding DFTAG\_SDD.

DFTAG\_SDL Scientific data labels  
? bytes  
704 (0x02C0)



*ref\_no* Reference number (16-bit integer)

*label\_n* Null terminated ASCII string (any length)

This tag points to a list of labels for the data in each dimension of the data set. Each label is a string terminated by a null byte (0).

DFTAG\_SDU Scientific data units  
? bytes  
705 (0x02C1)



*ref\_no* Reference number (16-bit integer)

*unit\_n* Null terminated ASCII string (any length)

This tag points to a list of strings specifying the units for the data and each dimension of the data set. Each unit's string is terminated by a null byte (0).

DFTAG\_SDF Scientific data format  
? bytes  
706 (0x02C2)



*ref\_no* Reference number (16-bit integer)

*format\_n* Null terminated ASCII string (any length)

This tag points to a list of strings specifying an output format for the data and each dimension of the data set. Each format string is terminated by a null byte (0).

DFTAG\_SDM Scientific data max/min  
8 bytes  
707 (0x02C3)



*ref\_no* Reference number (16-bit integer)

*max* Maximum value (type is specified by the data NT in the corresponding DFTAG\_SDD)

*min* Minimum value (type is specified by the data NT in the corresponding DFTAG\_SDD)

This record contains the maximum and minimum data values in the data set. The type of *max* and *min* are specified by the data NT of the corresponding DFTAG\_SDD.

DFTAG\_SDC Scientific data coordinates  
? bytes  
708 (0x02C4)



*ref\_no* Reference number (16-bit integer)

*string* Null terminated ASCII string (any length)

This tag points to a string specifying the coordinate system for the data set. The string is terminated by a null byte.

DFTAG\_SDLNK Scientific data set link  
8 bytes  
710 (0x02C6)



*ref\_no* Reference number (16-bit integer)

DFTAG\_NDG NDG tag (16-bit integer)

*NDG\_ref* NDG reference number (16-bit integer)

DFTAG\_SDG SDG tag (16-bit integer)

*SDG\_ref* SDG reference number (16-bit integer)

The purpose of this tag is to link together an old-style DFTAG\_SDG and a DFTAG\_NDG in cases where the NDG contains 32-bit floating point data and is, therefore, equivalent to an old SDG.

See also: Chapter , "Sets and Groups”

DFTAG\_CAL Calibration information  
36 bytes  
731 (0x02DB)



*ref\_no* Reference number (16-bit integer)

*cal* Calibration factor (64-bit IEEE float)

*cal\_err* Error in calibration factor (64-bit IEEE float)

*off* Calibration offset (64-bit IEEE float)

*off\_err* Error in calibration offset (64-bit IEEE float)

*data\_type* Constant representing the effective data type of the calibrated data (32-bit integer)

This tag points to a calibration record for the associated DFTAG\_SD. The data can be calibrated by first multiplying by the *cal* factor, then adding the *off* value. Also included in the record are errors for the calibration factor and offset and a constant indicating the effective data type of the calibrated data. Table 9P lists the available *data\_type* values.

* Available Calibrated Data Types

|  |  |
| --- | --- |
| Data Type | Description |
| DFTNT\_INT8 | Signed 8-bit integer |
| DFTNT\_UINT8 | Unsigned 8-bit integer |
| DFTNT\_INT16 | Signed 16-bit integer |
| DFTNT\_UINT16 | Unsigned 16-bit integer |
| DFTNT\_INT32 | Signed 32-bit integer |
| DFTNT\_UINT32 | Unsigned 32-bit integer |
| DFTNT\_FLOAT32 | 32-bit floating point |
| DFTNT\_FLOAT64 | 64-bit floating point |

DFTAG\_FV Fill value  
? bytes (size determined by size of data NT in corresponding  
 DFTAG\_SDD)  
732 (0x02DC)



*ref\_no* Reference number (16-bit integer)

*fill\_value* Value representing unset data in the corresponding DFTAG\_SD (size determined by size of data NT in corresponding DFTAG\_SDD)

This tag points to a value which has been used to indicate unset values in the associated DFTAG\_SD. The number type of the value (and, therefore, its size) is given in the corresponding DFTAG\_SDD.

### Vset Tags

DFTAG\_VG Vgroup  
14 + 4\**nelt* + *namelen* + *classlen* bytes   
1965 (0x07AD)



*ref\_no* Reference number (16-bit integer)

*nelt* Number of elements in the Vgroup (16-bit integer)

*tag\_n* Tag of the nth member of the Vgroup (16-bit integer)

*ref\_n* Reference number of the nth member of the Vgroup (16-bit integer)

*namelen* Length of the name field (16-bit integer)

*name* Non-null terminated ASCII string (length given by *namelen*)

*classlen* Length of the class field (16-bit integer)

*class* Non-null terminated ASCII string (length given by *classlen*)

*extag* Extension tag (16-bit integer)

*exref* Extension reference number (16-bit integer)

*version* Version number of DFTAG\_VG information (16-bit integer)

*more* Unused (2 zero bytes)

DFTAG\_VG provides a general-purpose grouping structure which can be used to impose a hierarchical structure on the tags in the group. Any HDF tag may be incorporated into a Vgroup, including other DFTAG\_VG tags.

See also: “Vsets, Vdatas, and Vgroups” in Chapter , "Sets and Groups”

*NCSA HDF Vsets, Version 2.0* for HDF Versions 3.2 and earlier

*HDF User’s Guide* and *HDF Reference Manual* for Versions 3.3 and 4.x

DFTAG\_VH Vdata description  
22 + 10\**nfields* + S*fldnmlen n* + *namelen* + *classlen* bytes  
1962 (0x07AA)



*ref\_no* Reference number (16-bit integer)

*interlace* Constant indicating interlace scheme used (16-bit integer)

*nvert* Number of entries in Vdata (32-bit integer)

*ivsize* Size of one Vdata entry (16-bit integer)

*nfields* Number of fields per entry in the Vdata (16-bit integer)

*type\_n* Constant indicating the data type of the nth field of the Vdata (16-bit integer)

*isize\_n* Size in bytes of the nth field of the Vdata (16-bit integer)

*offset\_n* Offset of the nth field within the Vdata (16-bit integer)

*order\_n* Order of the nth field of the Vdata (16-bit integer)

*fldnmlen\_n* Length of the nth field name string (16-bit integer)

*fldnm\_n* Non-null terminated ASCII string (length given by corresponding *fldnmlen\_n*)

*namelen* Length of the name field (16-bit integer)

*name* Non-null terminated ASCII string (length given by *namelen*)

*classlen* Length of the class field (16-bit integer)

*class* Non-null terminated ASCII string (length given by *classlen*)

*extag* Extension tag (16-bit integer)

*exref* Extension reference number (16-bit integer)

*version* Version number of DFTAG\_VH information (16-bit integer)

*more* Unused (2 zero bytes)

DFTAG\_VH provides all the information necessary to process a DFTAG\_VS.

See also: DFTAG\_VS (this section)

“Vsets, Vdatas, and Vgroups” in Chapter , "Sets and Groups”

*NCSA HDF Vsets, Version 2.0* for HDF Versions 3.2 and earlier

*HDF User’s Guide* and *HDF Reference Manual* for Versions 3.3 and 4.x

DFTAG\_VS Vdata  
 *nvert*, *isize\_n,* and *order\_n*  are specified in the   
 corresponding DFTAG\_VH  
1963 (0x07AB)



*ref\_no* Reference number (16-bit integer)

*vdata* Data block interpreted according to the corresponding DFTAG\_VH  
(value of the summation above, where *nvert,* *isize\_n,* and *order\_n*  are specified in the correspondingDFTAG\_VH)

DFTAG\_VS contains a block of data which is to be interpreted according to the information in the corresponding DFTAG\_VH.

See also: DFTAG\_VH (this section)

“Vsets, Vdatas, and Vgroups” in Chapter , "Sets and Groups”

*NCSA HDF Vsets, Version 2.0* for HDF Versions 3.2 and earlier

*HDF User’s Guide* and *HDF Reference Manual* for Versions 3.3 and 4.x

### Obsolete Tags

DFTAG\_ID8 Image dimension-8  
4 bytes  
200 (0x00C8)



*ref\_no* Reference number (16-bit integer)

*x\_dim* Length of x dimension (16-bit integer)

*y\_dim* Length of y dimension (16-bit integer)

The data for this tag consists of two 16-bit integers representing the width and height of an 8-bit raster image in bytes.

This tag has been superseded by DFTAG\_ID.

DFTAG\_IP8 Image palette-8  
768 bytes  
201 (0x00C9)



*ref\_no* Reference number (16-bit integer)

Table entries 256 triples of 8-bit integers

The data for this tag can be thought of as a table of 256 entries, each containing one value for red, green, and blue. The first triple is palette entry 0 and the last is palette entry 255.

This tag has been superseded by DFTAG\_LUT.

DFTAG\_RI8 Raster image-8  
*xdim\*ydim* bytes (where *xdim* and *ydim* are the dimensions   
 specified in the corresponding DFTAG\_ID8)  
202 (0x00CA)



*ref\_no* Reference number (16-bit integer)

Image data 2-dimensional array of 8-bit integers

The data for this tag is a row-wise representation of the elementary 8-bit image data. The data is stored width-first (i.e., row-wise) and is 8 bits per pixel. The first byte of data represents the pixel in the upper-left hand corner of the image.

This tag has been superseded by DFTAG\_RI.

DFTAG\_CI8 Compressed image-8  
? bytes  
203 (0x00CB)



*ref\_no* Reference number (16-bit integer)

*compressed\_image* Series of run-length encoded bytes

The data for this tag is a row-wise representation of the elementary 8-bit image data. Each row is compressed using the following run-length encoding where *n* is the lower seven bits of the byte. The high bit indicates whether the following *n* bytes will be reproduced exactly (high bit = 0) or whether the following byte will be reproduced *n* times (high bit = 1). Since DFTAG\_CI8 and DFTAG\_RI8 are basically interchangeable, it is suggested that you not have a DFTAG\_CI8 and a DFTAG\_RI8 with the same reference number.

This tag has been superseded by DFTAG\_RLE.

DFTAG\_II8 IMCOMP image-8  
? bytes  
204 (0x00CC)



*ref\_no* Reference number (16-bit integer)

*compressed\_image*   
Compressed image data

The data for this tag is a 4:1 compressed 8-bit image, using the IMCOMP compression scheme.

This tag has been superseded by DFTAG\_IMC.

DFTAG\_SDG Scientific data group  
*n*\*4 bytes (where *n* is the number of data objects in the group)  
700 (0x02BC)



*ref\_no* Reference number (16-bit integer)

*tag\_n* Tag number of nth member of the group (16-bit integer)

*ref\_n* Reference number of nth member of the group (16-bit integer)

The SDG data element contains a list of tag/refs that define a scientific data set. All of the members of the group provide information required to correctly interpret and display the data. Application programs that deal with SDGs should read all of the elements of an SDG and process those which it can use. Even if an application cannot process all of the objects, the objects that it can understand will be usable.

Table 9Q lists the tags that may appear in an SDG.

* Available SDG Tags

|  |  |
| --- | --- |
| Tag | Description |
| DFTAG\_SDD | Scientific data dimension record (rank and dimensions) |
| DFTAG\_SD | Scientific data |
| DFTAG\_SDS | Scales |
| DFTAG\_SDL | Labels |
| DFTAG\_SDU | Units |
| DFTAG\_SDF | Formats |
| DFTAG\_SDM | Maximum and minimum values |
| DFTAG\_SDC | Coordinate system |
| DFTAG\_SDT | Transposition (obsolete) |
| DFTAG\_SDLNK | Link to new DFTAG\_NDG |

Example

DFTAG\_SDD, DFTAG\_SD, DFTAG\_SDM

Assume that a dimension record, scientific data, and the maximum and minimum values of the data are required to read and interpret a particular data set. These data objects can be associated in an SDG so that an application can read the rank and dimensions from the dimension record and then read the data array. If the application needs the maximum and minimum values, it will read them as well.

This tag has been superseded by DFTAG\_NDG**.**

See also: Chapter , "Sets and Groups”

DFTAG\_SDT Scientific data transpose  
0 bytes  
709 (0x02C5)



*ref\_no* Reference number (16-bit integer)

The presence of this tag in a group indicates that the data pointed to by the corresponding DFTAG\_SD is in column-major order, instead of the default row-major order. No data is associated with this tag.

This tag is no longer written by the HDF library. When it is encountered in an old file, it is interpreted as originally intended.

1. The document *JPEG File Interchange Format* has not been published again since its latest version v1.02, on September 1, 1992. An electronic copy is available at http://www.w3.org/Graphics/JPEG/jfif3.pdf. [↑](#footnote-ref-1)