

# TIFF (Tag Image File Format): Specifications (4 of 7)

Revised: 3/22/88
Security: Everyone

TIFF (Tag Image File Format): Specifications (4 of 7)

==========

This article last reviewed: 12 February 1988

5. The Fields, continued
---

StripByteCounts
Tag = 279 (117)
Type = LONG
N = StripsPerImage for PlanarConfiguration equal to 1.
= SamplesPerPixel * StripsPerImage for PlanarConfiguration equal to 2

For each strip, the number of bytes in that strip.
No default.

SamplesPerPixel
Tag = 277 (115)
Type = SHORT
N = 1

The number of samples per pixel. Usually 1 for monochromatic data and 3 for color data (i.e. one sample for each of the color planes.)
Default = 1.

BitsPerSample
Tag = 258 (102)
Type = SHORT
N = SamplesPerPixel

Number of bits per sample. Note that this tag allows a different number of bits per sample for each sample corresponding to a pixel. For example, RGB color data could use a different number of bits per sample for each of the three color planes.
Default = 1.

PlanarConfiguration
Tag = 284  (11C)
Type = SHORT
N = 1

1 = the sample values for each pixel are stored contiguously, so that there is a single image plane. See PhotometricInterpretation to determine the order of the samples within the pixel data.
2 = the samples are stored in separate "sample planes." The values in StripOffsets and StripByteCounts are then arranged as a 2-dimensional array, with SamplesPerPixel rows and StripsPerImage columns. (All of the columns for row 0 are stored first, followed by the columns of row 1, and so on.) PhotometricInterpretation describes the type of data that is stored in each sample plane.

If SamplesPerPixel is 1, a PlanarConfiguration value of 1 is equivalent to a value of 2.
No default.

Compression
Tag = 259  (103)
Type= SHORT
N= SamplesPerPixel for PlanarConfiguration equal to 1 or 2.

Note that a value is provided for each sample, allowing different compression schemes to be applied to different planes of data.

1 = No compression, but pack data into bytes as tightly as possible, with no unused bits except at the end of a row. See also FillOrder. The bytes are stored as an array of type BYTE, for BitsPerSample <= 8, SHORT if BitsPerSample > 8 and <= 16, and LONG if BitsPerSample > 16 and <= 32. The byte ordering of data >8 bits must be consistent with that specified in the TIFF file header (bytes 0 and 1). "Intel" format files will have the least significant bytes preceeding the most significant bytes while "Motorola" format files will have the opposite.

If the number of bits per sample is not a power of 2, and you are willing to give up some space for better performance, you may wish to use the next higher power of 2. For example, if your data can be represented in 6 bits, you may wish to specify that it is 8 bits deep. If you take this approach, you should be sure that MinSampleValue and MaxSampleValue are given correct values (probably 0 and 63 for intrinsically 6-bit data.) TIFF file readers should use MinSampleValue and MaxSampleValue to determine the range of values in the data rather than BitsPerSample.

Rows are required to begin on byte boundaries.

2 = CCITT Group 3 1-Dimensional Modified Huffman run length encoding. BitsPerSample must be 1, since this type of compression is defined only for "binary" images.

3 = Facsimile-compatible CCITT Group 3, exactly as specified in "Standardization of Group 3 facsimile apparatus for document transmission," Recommendation T.4, Volume VII, Fascicle VII.3, Terminal Equipment and Protocols for Telematic Services, The International Telegraph and Telephone Consultative Committee (CCITT), Geneva, 1985, pages 16 through 31. Each strip must begin on a byte boundary. (But recall that an image can be a single strip.) Rows that are not the first row of a strip are not required to begin on a byte boundary. The data is stored as bytes, not words -- byte-reversal is not allowed. Note that the FillOrder field still applies. See the Group3Options field for Group 3 options such as 1D vs 2D coding.

4 = Facsimile-compatible CCITT Group 4, exactly as specified in "Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Apparatus," Recommendation T.6, Volume VII, Fascicle VII.3, Terminal Equipment and Protocols for Telematic Services, The International Telegraph and Telephone Consultative Committee (CCITT), Geneva, 1985, pages 40 through 48. Each strip must begin on a byte boundary. Rows that are not the first row of a strip are not required to begin on a byte boundary. The data is stored as bytes, not words. Note that the FillOrder field still applies. See the Group4Options field for Group 4 options.

32771 = the same thing as Compression type 1 (no compression), except that each row begins on the next available word boundary, instead of byte boundary.

32773 = PackBits compression, a relatively simple byte-oriented run-length scheme.

Data compression only applies to pixel data, as pointed to by StripOffsets. All other TIFF information is unaffected.

To be determined are additional compression schemes for gray and colored images. We encourage your suggestions, especially if accompanied by full specifications and performance information. It is of course desirable to minimize the number of compression schemes that are being used, but this is clearly an area in which extremely significant time and space tradeoffs exist.
Default = 1.

Group3Options
Tag = 292 (124)
Type = LONG
N = 1

This field is made up of a set of 32 flag bits. Unused bits are expected to be 0. Bit 0 is the low-order bit. It is probably not safe to try to read the file if any bit of this field is set that you don't know the meaning of.

Bit 0 is 1 for 2-dimensional coding (else 1-dimensional is assumed). For 2-D coding, if more than one strip is specified, each strip must begin with a 1-dimensionally coded line. That is, RowsPerStrip should be a multiple

of "Parameter K" as documented in the CCITT specification.

Bit 1 is 1 if uncompressed mode is used.

Bit 2 is 1 if fill bits have been added as necessary before EOL codes such that EOL always ends on a byte boundary, thus ensuring an eol-sequence of a 1 byte preceded by a zero nibble:  xxxx-0000 0000-0001.

Default is 0, for basic 1-dimensional coding.

Group4Options
Tag =  293  (125)
Type = LONG
N = 1

This field is made up of a set of 32 flag bits.  Unused bits are expected to be 0.  Bit 0 is the low-order bit.  It is probably not safe to try to read the file if any bit of this field is set that you don't know the meaning of. Gray scale and color coding schemes are under study, and will be added when finalized.

For 2-D coding, each strip is encoded as if it were a separate image. In particular, each strip begins on a byte boundary; and the coding for the first row of a strip is encoded independently of the previous row, using horizontal codes, as if the previous row is entirely white.  Each strip ends with the 24-bit end-of-facsimile block (EOFB).

Bit 0 is unused.

Bit 1 is 1 if uncompressed mode is used.

Default is 0, for basic 2-dimensional binary compression.

FillOrder
Tag = 266  (10A)
Type = SHORT
N = 1

The order of data values within a byte.
1 = most significant bits of the byte are filled first. That is, data values (or code words) are ordered from high order bit to low order bit within a byte.
2 = least significant bits are filled first.
Default is FillOrder = 1.



Tech Info Library Article Number:349