

Table 16: Valid TDISPn format values in TABLE extensions.

Field value	Data type
Aw	Character
Iw.m	Integer
Bw.m	Binary, integers only
Ow.m	Octal, integers only
Zw.m	Hexadecimal, integers only
Fw.d	Floating-point, fixed decimal notation
Ew.dEe	Floating-point, exponential notation
ENw.d	Engineering; E format with exponent multiple of three
ESw.d	Scientific; same as EN but non-zero leading digit if not zero
Gw.dEe	General; appears as F if significance not lost, else E.
Dw.dEe	Floating-point, exponential notation

Notes. *w* is the width in characters of displayed values, *m* is the minimum number of digits displayed, *d* is the number of digits to right of decimal, and *e* is number of digits in exponent. The *.m* and *Ee* fields are *optional*.

erators). String comparisons with the TTYPEn keyword values *should not* be case sensitive (e.g., 'TIME' and 'Time' *should* be interpreted as the same name).

It should be noted that under the Green Bank convention (Section 8.2.1) a keyword may be represented as a field in a table, where the field name, as given by the TTYPEn keyword, equals the keyword name. This convention is commonly used in cases where the same value of the keyword does not apply to every row of the table.

It is *strongly recommended* that if the value of a TTYPEn keyword is the same as the name of a reserved keyword, than the quantity in that field should conform to the definition of that keyword.

TUNITn keywords. The value field *shall* contain a character string describing the physical units in which the quantity in field *n*, after any application of TSCALn and TZEROn, is expressed. Units *must* follow the prescriptions in Sect. 4.3.

TSCALn keywords. This indexed keyword *shall* be used, along with the TZEROn keyword, to linearly scale the values in the table field *n* to transform them into the physical values that they represent using Eq. 7. The value field *shall* contain a floating-point number representing the coefficient of the linear term in the scaling equation. The default value for this keyword is 1.0. This keyword *must not* be used for A-format fields.

The transformation equation used to compute a true physical value from the quantity in field *n* is

$$\text{physical_value} = \text{TZEROn} + \text{TSCALn} \times \text{field_value} \quad (7)$$

where *field_value* is the value that is actually stored in that table field in the FITS file.

TZEROn keywords. This indexed keyword *shall* be used, along with the TSCALn keyword, to linearly scale the values in the table field *n* to transform them into the physical values that they represent using Eq. 7. The value field *shall* contain a floating-point number representing the physical value corresponding to an array value of zero. The default value for this keyword is 0.0. This keyword *must not* be used for A-format fields.

Table 17: Mandatory keywords in binary table extensions.

#	Keyword
1	XTENSION= <u> </u> 'BINTABLE'
2	BITPIX = 8
3	NAXIS = 2
4	NAXIS1
5	NAXIS2
6	PCOUNT
7	GCOUNT = 1
8	TFIELDS
	:
	(other keywords, including (if TFIELDS is not zero) ...)
	TTYPEn, n=1, 2, ..., k where k is the value of TFIELDS (<i>Recommended</i>)
	TFORMn, n=1, 2, ..., k where k is the value of TFIELDS (<i>Required</i>)
	:
last	END

Table 18: Valid TFORMn data types in BINTABLE extensions.

TFORMn value	Description	8-bit Bytes
L	Logical	1
X	Bit	†
B	Unsigned byte	1
I	16-bit integer	2
J	32-bit integer	4
K	64-bit integer	8
A	Character	1
E	Single precision floating point	4
D	Double precision floating point	8
C	Single precision complex	8
M	Double precision complex	16
P	Array Descriptor (32-bit)	8
Q	Array Descriptor (64-bit)	16

Notes. ^(†) Number of 8-bit bytes needed to contain all bits.

95, hexadecimal 5F) character. Use of other characters is *not recommended* because it may be difficult to map the column names into variables in some languages (e.g., any hyphens, '*' or '+' characters in the name may be confused with mathematical operators). String comparisons with the TTYPEn keyword values *should not* be case sensitive (e.g., 'TIME' and 'Time' *should* be interpreted as the same name).

It should be noted that under the Green Bank convention (Section 8.2.1) a keyword may be represented as a field in a table, where the field name, as given by the TTYPEn keyword, equals the keyword name. This convention is commonly used in cases where the same value of the keyword does not apply to every row of the table.

It is *strongly recommended* that if the value of a TTYPEn keyword is the same as the name of a reserved keyword, than the quantity in that field should conform to the definition of that keyword.

TUNITn keywords. The value field *shall* contain a character string describing the physical units in which the quantity in field n, after any application of TSCALn and TZEROn, is expressed. Units *must* follow the prescriptions in Sect. 4.3.

TSCALn keywords. This indexed keyword *shall* be used, along with the TZEROn keyword, to linearly scale the values in the table field n to transform them into the physical values that they represent using Eq. 7. It *must not* be used if the format of field n is A, L, or X. For fields with all other data types, the value field *shall* contain a floating-point number representing the coefficient of the linear term in Eq. 7, which is used to compute the true physical value of the field, or, in the case of the complex data types C and M, of the real part of the field, with the imaginary part of the scaling factor set to zero. The default value for this keyword is 1.0. For fields of type P or Q, the values of TSCALn and TZEROn are to be applied to the values in the data array in the heap area, not the values of the array descriptor (see Sect. 7.3.5).

TZEROn keywords. This indexed keyword *shall* be used, along with the TSCALn keyword, to linearly scale the values in the table field n to transform them into the physical values that they represent using Eq. 7. It *must not* be used if the format of field n is A, L, or X. For fields with all other data types, the value field *shall* contain a floating-point number representing the true physical value corresponding to a value of zero in field n of the FITS file, or, in the case of the complex data types C and M, in the real part of the field, with the imaginary part set to zero. The default value for this keyword is 0.0. Equation 7 is used to compute a true physical value from the quantity in field n. For fields of type P or Q, the values of TSCALn and TZEROn are to be applied to the values in the data array in the heap area, not the values of the array descriptor (see Sect. 7.3.5).

In addition to its use in representing floating-point values as scaled integers, the TZEROn keyword is also used when storing unsigned integer values in the field. In this special case the TSCALn keyword *shall* have the default value of 1.0 and the TZEROn keyword *shall* have one of the integer values shown in Table 19

Since the binary table format does not support a native unsigned integer data type (except for the unsigned 8-bit 'B' column type), the unsigned values are stored in the field as native signed integers with the appropriate integer offset specified by the TZEROn keyword value shown in the table. For the byte column type, the converse technique can be used to store signed byte values as native unsigned values with the negative TZEROn offset. In each case, the physical value is computed by adding the

Some non-linear algorithms that describe the transformation between pixel and intermediate coordinate axes require parameter values. A few non-linear algorithms also require character-valued parameters, e.g., table lookups require the names of the table extension and the columns to be used. Where necessary parameter values *must* be specified via the following keywords:

PVi_m – [floating point]. Numeric parameter values for intermediate world coordinate axis *i*, where *m* is the parameter number. Leading zeros *must not* be used, and *m* may have only values in the range 0 through 99, and that are defined for the particular non-linear algorithm.

PSi_m – [character]. Character-valued parameters for intermediate world coordinate axis *i*, where *m* is the parameter number. Leading zeros *must not* be used, and *m* may have only values in the range 0 through 99, and that are defined for the particular non-linear algorithm.

The following keywords, while not essential for a complete specification of an image WCS, can be extremely useful for readers to interpret the accuracy of the WCS representation of the image.

CRDER_i – [floating point; default: 0.0]. Random error in coordinate *i*, which *must* be non-negative.

CSYER_i – [floating point; default: 0.0]. Systematic error in coordinate *i*, which *must* be non-negative.

These values *should* give a representative average value of the error over the range of the coordinate in the HDU. The total error in the coordinates would be given by summing the individual errors in quadrature.

8.2.1. Green Bank convention

In cases where multiple images are stored within a vector column of a *FITS* binary table (where each row of the table contains a different image), it also may be necessary to record the WCS keyword values for each image. Since, in general, each image may have different values for some of the WCS parameters, a single set of WCS keywords in the table header is insufficient for recording the values for every image. In this case, the WCS information for each image may be stored in the table by expanding any WCS keyword that does not have the same value in every image into a table field that has the same name as the keyword. Each row of that column can then contain the WCS parameter value that applies to the corresponding vector column image in that same row.

For example, if each image in the table has different **CRVAL1** and **CRVAL2** keyword values, then 2 floating-point data type columns can be added to the table with **TTYPEn** = 'CRVAL1' and **TTYPEn** = 'CRVAL2' to record the values of those parameters for each image. Note that, in principle, it would be more precise to use the binary table vector form of these WCS keywords as shown in Table 22 (e.g., **iCRVLn** instead of **CRVALi** for the column names, but if there is only one vector column in the table, and thus there is no ambiguity about which image the WCS parameter refers to, then the 'Primary array' form of the parameter name may be used instead. (This is also allowed because this convention pre-dates the creation of the alternate forms of the WCS keywords). In the case of WCS parameters that have the same value for every image, there is no need to expand them

into table columns, and they can be expressed normally using the single standard WCS header keywords.

This general concept of expanding a keyword into a table column is generally known as the Green Bank¹⁰ keyword convention. For example, if the information given in each row of an ASCII or binary table corresponds to a different date, then the table could have a column with **TTYPEn** = 'DATE-OBS' to store the date value for each row. Similarly, if all the values in a column contain the same value, then that column could be collapsed into a keyword with the same name as the previous column.

8.2.2. Alternative WCS axis descriptions

In some cases it is useful to describe an image with more than one coordinate type¹¹. Alternative WCS descriptions *may* be added to the header by adding the appropriate sets of WCS keywords, and appending to all keywords in each set an alphabetic code in the range A through Z. Keywords that may be used in this way to specify a coordinate system version are indicated in Table 22 with the suffix *a*. All implied keywords with this encoding are *reserved keywords*, and *must* only be used in *FITS* HDUs as specified in this Standard. The axis numbers *must* lie in the range 1 through 99, and the coordinate parameter *m* *must* lie in the range 0 through 99, both with no leading zeros.

The *primary* version of the WCS description is that specified with *a* as the blank character¹². Alternative axis descriptions are optional, but *must not* be specified unless the primary WCS description is also specified. If an alternative WCS description is specified, all coordinate keywords for that version *must* be given even if the values do not differ from those of the primary version. Rules for the default values of alternative coordinate descriptions are the same as those for the primary description. The alternative descriptions are computed in the same fashion as the primary coordinates. The type of coordinate depends on the value of **CTYPEia**, and may be linear in one of the alternative descriptions and non-linear in another.

The alternative version codes are selected by the *FITS* writer; there is no requirement that the codes be used in alphabetic sequence, nor that one coordinate version differ in its parameter values from another. An optional keyword **WCSNAMEa** is also defined to name, and otherwise document, the various versions of WCS descriptions:

WCSNAMEa – [character; default for *a*: '␣' (i.e., blank, for the primary WCS, else a character A through Z that specifies the coordinate version)]. Name of the world coordinate system represented by the WCS keywords with the suffix *a*. Its primary function is to provide a means by which to specify a particular WCS if multiple versions are defined in the HDU.

¹⁰ Named after a meeting held in Green Bank, West Virginia, USA in 1989 to develop standards for the interchange of single dish radio astronomy data.

¹¹ Examples include the frequency, velocity, and wavelength along a spectral axis (only one of which, of course, could be linear), or the position along an imaging detector in both meters and degrees on the sky.

¹² There are a number of keywords (e.g. **ijPCna**) where the *a* could be pushed off the 8-char keyword name for plausible values of *i*, *j*, *k*, *n*, and *m*. In such cases *a* is still said to be 'blank' although it is not the blank character.

H.3 List of modifications to the latest FITS standard

The Green Bank convention described in Sect. 8.2.1 was in use since 1989, and registered in 2010. The text of the registered convention is reported at <http://fits.gsfc.nasa.gov/registry/greenbank/greenbank.pdf>. The differences with this standard concern:

- The original convention made reference only to binary tables, while the present standard applies it as well to ASCII tables.
- The original text contained some additional details about the history of the convention.

DRAFT