The *FITS* IMAGE Extension.
A Proposal

J.D. Ponz (ESA), R.W. Thompson (CSC), J.R. Muñoz (ESA)
TN/8017-01/JDP/920207

7 February 1992

Abstract

This paper describes the *FITS* IMAGE extension which provides a simple means for storing two or more related but not necessarily identical arrays in a single *FITS* file. The proposed extension is structured in a manner identical to that of a standard *FITS* header and data unit, thereby making it simple to interpret and if desired, easily converted to separate *FITS* primary array files.

1 Introduction

The IMAGE extension conforms to the generalized *FITS* extension format described by Grosbøl et al (ref. 3). It was first proposed by Muñoz (ref. 6) for use by the International Ultraviolet Explorer (IUE) project, as a means for combining image data with related auxiliary information which could not be easily stored using GROUP format or merged with the image data to create a single primary array. The IMAGE extension is a simple replication of the primary data array with the following features:

1. Allows the storage of an unlimited number of multidimensional arrays. (Note: support for multidimensional arrays has been suggested for, but is not currently included in, the proposed binary table extension (ref. 7).
2. Arrays are stored in separate extensions, thereby allowing each array to have its own header and semantic contents.
3. *FITS* readers can easily skip over reading separate extensions.
4. No additional keywords or conventions need to be adopted.
5. Dividing each individual header and data unit in a *FITS* IMAGE extension file results in simple primary array format files without any further processing (other than replacing the `XTENSION = 'IMAGE␣␣␣'` keyword with `SIMPLE = T`).

The IMAGE extension is being proposed for use in the IUE Archive reprocessing project, to store data quality flags associated with the spectral images.

2 Identification

The IMAGE extension may appear in a *FITS* file if the primary header of the *FITS* file has the keyword `EXTEND` set to `T` and the first keyword of the associated extension header has `XTENSION = 'IMAGE␣␣␣'`.

No other modifications are required although `NAXIS = 0` is not recommended since it would not make sense to extend a non-existing image with another image.

3 Extension Header

The extension header would begin in the first byte of the first record following the primary data array or the previous extension file. The format is identical to the primary header format and is described in the *FITS* Draft standard (ref. 7). The content of the header should be sufficient to allow a *FITS* reader to decide if it should read or skip over the extension.

3.1 Required Keywords

The card images in the header of the IMAGE extension use the following keywords in the order indicated in Table 1.

Principal HDU	Proposed IMAGE Extension
`SIMPLE`	`XTENSION`[1]
`BITPIX`	`BITPIX`
`NAXIS`	`NAXIS`
`NAXISn`	`NAXISn`[2]
`EXTEND`[3]	`PCOUNT = 0`
`END`	`GCOUNT = 1`
	`END`

[1] `XTENSION=␣'IMAGE␣␣␣'` for the proposed extension.
[2] `n = 1, ..., NAXIS`.
[3] Required only if extension present.

Table 1: Mandatory *FITS* keywords for the Principal HDU and the proposed IMAGE extension.

3.1.1 `XTENSION` Keyword

The value field shall contain the character string `'IMAGE␣␣␣'`.

3.1.2 `BITPIX` Keyword

The value field shall contain an integer. The absolute value is used in computing the size of the data structure. It shall specify the number of bits that represent a data value. All `BITPIX` values allowed in the Primary Header for the primary data matrix are also allowed for the IMAGE extension header.

3.1.3 `NAXIS` Keyword

The value field shall contain a non-negative integer no greater than 999, representing the number of axes in the data array.

3.1.4 `NAXISn` Keywords

The value field of this indexed keyword shall contain a non-negative integer, representing the number of positions along axis `n` (`n= 1, 2, ..., NAXIS`) of the data array.

3.1.5 `PCOUNT` Keyword

The value field shall contain a non-negative integer that shall be used in any way appropriate to define the data structure, consistent with the equation for `NBITS` in the *FITS* Draft Standard (ref. 7). A simple IMAGE extension will have `PCOUNT = 0`.

3.1.6 `GCOUNT` Keyword

The value field shall contain a non-negative integer that shall be used in any way appropriate to define the data structure, consistent with the equation for `NBITS` in the *FITS* Draft Standard (ref. 7). A simple IMAGE extension will have `GCOUNT = 1`.

3.1.7 `END` Keyword

This keyword has no associated value. Columns 9-80 shall be filled with ASCII blanks.

Conforming Extension	Bibliographic Keywords	Commentary Keywords	Observation Keywords	Array Keywords
`EXTNAME`	`AUTHOR`	`COMMENT`	`DATE-OBS`	`BSCALE`
`EXTVER`	`REFERENC`	`HISTORY`	`TELESCOP`	`BZERO`
`EXTLEVEL`		␣␣␣␣␣␣␣␣	`INSTRUME`	`BUNIT`
			`OBSERVER`	`BLANK`
			`OBJECT`	`CTYPEn`
			`EQUINOX`	`CRPIXn`
			`EPOCH`[1]	`CROTAn`
				`CRVALn`
				`CDELTn`
				`DATAMAX`
				`DATAMIN`

[1] See comment on `EPOCH` in ref. 7.

Table 2: Reserved keywords for the proposed IMAGE extension. The keywords are defined in the *FITS* Draft Standard (ref.7).

3.2 Reserved Keywords

Additional keywords, located in the extension header after the `GCOUNT` keyword and before `END`, may be used to describe history of the data, characteristics of the observations, characteristics of the data array and other information.

These keywords are optional but, when used to describe an extension, they shall have the meaning described in the *FITS* Draft Standard (ref. 7). Table 2 summarizes the reserved keywords.

4 Data Sequence

The data format is identical to that of a primary data array and is described in detail in the *FITS* Draft Standard (ref. 7). This format will allow each IMAGE extension to contain a single data array of 1-999 dimensions with a data structure and scale factors independent of other arrays.

The IMAGE extension data shall begin in the first byte of the first record following the last header record. The first value of each subsequent row of the array shall immediately follow the last value of the previous row. In this manner, the array structure is independent of the record structure. Arrays of more than one dimension shall consist of a sequence such that the index of the first dimension varies most rapidly, and the index of the last dimension varies least rapidly.

5 Example IMAGE Extension Header

The following is an example of how this type of extension can be used. In this example, the primary data array contains an IUE linearized image file with the associated pixel quality flags stored using the IMAGE extension.

Main Header

```
         1         2         3         4         5         6         7
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345...
--------------------------------------------------------------------------------
SIMPLE  =                    T / Standard FITS format
BITPIX  =                   16 / 2-Bytes, 2-s complement integers
NAXIS   =                    2 / Number of axes
NAXIS1  =                  768 / Number of pixels per row
NAXIS2  =                  768 / Number of rows
EXTEND  =                    T / Extensions may be present
CTYPE1  = 'SAMPLE  '           / X axis
CTYPE2  = 'LINE    '           / Y axis
BSCALE  =           3.1250E-02 / REAL = (FITS * BSCALE) + BZERO
BZERO   =                   0. / Bias
ORIGIN  = 'VILSPA  '           / Institution generating tape
TELESCOP= 'IUE     '           / IUE telescope
FILENAME= 'SWP12345.LIHI'      / Filename (camera)(image).LI(disp)
DATE    = '12/10/92'           / Date tape was written as DD/MM/YY
...
END
```

Extension Header

```
         1         2         3         4         5         6         7
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345...
--------------------------------------------------------------------------------
XTENSION= 'IMAGE   '           / IMAGE extension
BITPIX  =                   16 / 2-Bytes, 2-s complement integers
NAXIS   =                    2 / Number of axes
NAXIS1  =                  768 / Number of pixels per row
NAXIS2  =                  768 / Number of rows
PCOUNT  =                    0 / Number of parameters per group
GCOUNT  =                    1 / Number of groups
CTYPE1  = 'SAMPLE  '           / X axis
CTYPE2  = 'LINE    '           / Y axis
FILENAME= 'SWP12345.LFHI'      / Filename (camera)(image).LF(disp)
EXTNAME = 'LFHI    '           / Data quality flags
...
END
```

6 References

1. Wells, D.C., Greisen, E.W., Harten, R.H. 1981, "*FITS*: A Flexible Image Transport System", *Astron. Astrophys. Suppl.*, **44**, 363-370.

2. Greisen, E.W., Harten, R.H. 1981, "An Extension of *FITS* for Small Arrays of Data", *Astron. Astrophys. Suppl.*, **44**, 371-374.

3. Grosbøl, P., Harten, R.H., Greisen, E.W., Wells, D.C. 1988, "Generalized Extensions and Blocking Factors for *FITS*", *Astron. Astrophys. Suppl.*, **73**, 359-364.

4. Harten, R.H., Grosbøl, P., Greisen, E.W., Wells, D.C. 1988, "The *FITS* Table Extension", *Astron. Astrophys. Suppl.*, **73**, 365-372.

5. Wells, E. W., Grosbøl, P. 1990, "Floating Point Agreement for *FITS*", (available from the NOST *FITS* Support Office).

6. Muñoz, J.R. 1989, "IUE Data in *FITS* Format", ESA IUE Newsletter, **32**, 12-45.

7. NOST 1991, "Implementation of the Flexible Image Transport System", NOST 100-0.3b Draft Standard. NASA/OSSA Office of Standards and Technology, Goddard Space Flight Center.