SCHOOL OF PHYSICS AND ASTRONOMY
UNIVERSITY OF BIRMINGHAM
DEV/003

Asterix Data Analysis
Development Note 003

Richard Beard
June 23, 1998

Extending FITS I/O within ASTERIX

Contents

1 Introduction

The current version of ASTERIX (2.2) has the capability to read and write data in both **H**ierarchical **D**ata **S**tructure (HDS) and **F**lexible **I**mage **T**ransport **S**ystem (FITS) formats. However, this duality is not universal, and the FITS compatibility applies only to certain applications.

More specifically FITS I/O is available with Isys, graphical display, image processing and general utilities. It is not available with ROSAT data analysis, source searching, spectral fitting or time series analysis.

2 HDS vs. FITS

These two data formats are quite distinct from one another in a number of ways;

	HDS	**FITS**
File size:	Large datasets approx. 2x FITS	Smaller, but short files larger than HDS due to 2880 byte blocking
Storage:	Hierarchical tree structure	Linear data blocks
Access:	Pointer to each structure object	By reference through block number
Identifiers:	15 characters max.	8 characters max.

Such differences make it difficult to manipulate the same data equally in both formats. For example, a HDS structure can have any number of data arrays within it, whereas a FITS block may only store one object (unless a binary table is used, making it difficult to add and delete objects once created).

3 FITS and Source Searching

An initial attempt has been made to make the source searching system compatible with FITS files, though the progress is rather limited so far. See the directory `/lsoft1/asterix/work/rb/psf` for more details. Here an attempt is made to attach spatial response file to a FITS image with the application `SPRESP`.

An extra set of methods has been created (see Programming Note 17) for dealing with the response. This work assumed that the FITS format was at least slightly compatible with original program structure, which may not in fact be the case. The work is closely linked to the proceeding section, which is highly developmental, and thus is likely to be of little final use.

See the selection of OGIP documents stored at HEASARC for more information on other proposed ways of storing searching/instrument data within a FITS file.

4 A Possible Extension to FITS

One way to make the FITS format as flexible as HDS is by introducing some system for the linking together of data extension. This can be done by adding extra KEYWORDS into each data block that define a tree-like structure within the existing linear arrangement.

Like many such FITS extensions, whilst useful, there would be little chance of having the proposal adopted by OGIP, and thus any use made of this system by ASTERIX might limit its use to ASTERIX created FITS files.

The following extra keywords are proposed;

`PARENT`	The parent of this block in the tree structure. If missing block is in the top level.
`CHILDREN`	A count of the number of leaf blocks to this node in the tree. If missing the block has no children.
`CHILDnnn`	The block name of the `nnn`'th leaf to this node.

Block names are those defined by the `EXTNAME` keyword.

If the reading application can parse these extra keywords then the implied hierarchical structure can be unfolded, otherwise the existing linear structure will be unaffected. Code to implement this scheme is available in `/lsoft1/asterix/work/rb/adi/newsrc`.

4.1 Example

This example shows a simple file structure as traced out with `HTRACE` and the equivalent keywords from the FITS file. It comes from an attempt to develop an `FTRACE` command, the FITS equivalent of `HTRACE`; see the directory `/lsoft1/asterix/work/rb/fed`

4.1.1 HDS file

```
HTRACE Version 2.2-0
 FOO              <UNKNOWN>      {structure}
   DATA_ARRAY[2,3]  <_REAL>        1.23456,1.23456,1.23456,1.23456,1.23456,1.23456

   MORE             <EXT>          {structure}
     ASTERIX          <EXT>          {structure}
   HISTORY          <HISTORY>      {structure}

     CREATED          <_CHAR*18>     "12-Feb-98 14:38:06"
     UPDATE_MODE      <_CHAR*10>     "NORMAL    "
     EXTEND_SIZE      <_INTEGER>     10
     CURRENT_RECORD   <_INTEGER>     1
     RECORDS[10]      <HIST_REC>     {array of structures}

     Contents of RECORDS[1]
       HOST             <_CHAR*30>     "xund                          "
       DATE             <_CHAR*18>     "12-Feb-98 14:38:06"
       COMMAND          <_CHAR*30>     "WRITE1 V1.0                   "
```

4.1.2 FITS file

Most of the keywords have been deleted to make the example shorter.

```
SIMPLE  =                    T / file does conform to FITS standard
BITPIX  =                    8 / bits per pixel
NAXIS   =                    0 / number of data axes
EXTEND  =                    T / FITS dataset may contain extensions
COMMENT   FITS (Flexible Image Transport System) format defined in Astronomy and
COMMENT   Astrophysics Supplement Series v44/p363, v44/p371, v73/p359, v73/p365.
COMMENT   Contact the NASA Science Office of Standards and Technology for the
COMMENT   FITS Definition document #100 and other FITS information.
CHILDREN=                    3 / Number of children for this HDU
CHILD1  = 'DATA_ARRAY'         / Name of child extension
CHILD2  = 'MORE    '           / Name of child extension
CHILD3  = 'HISTORY '           / Name of child extension
CREATOR = 'WRITE1 V1.0'        / Creator of this file
END

XTENSION= 'IMAGE   '           / IMAGE extension
BITPIX  =                  -32 / bits per pixel
NAXIS   =                    2 / number of data axes
NAXIS1  =                    2 / length of data axis 1
NAXIS2  =                    3 / length of data axis 2
PCOUNT  =                    0 / number of random group parameters
GCOUNT  =                    1 / number of random groups
EXTNAME = 'DATA_ARRAY'         / Extension contains DATA_ARRAY structure
TYPE    = 'REAL    '           / Data type of this extension
PARENT  = 'PRIMARY '           / Parent HDU of this extension
CHILDREN=                    0 / Number of children for this HDU
END

XTENSION= 'IMAGE   '           / IMAGE extension
...
EXTNAME = 'MORE    '           / Extension contains MORE structure
TYPE    = 'EXT     '           / Data type of this extension
PARENT  = 'PRIMARY '           / Parent HDU of this extension
CHILDREN=                    1 / Number of children for this HDU
CHILD1  = 'ASTERIX '           / Name of child extension
END

XTENSION= 'IMAGE   '           / IMAGE extension
...
EXTNAME = 'ASTERIX '           / Extension contains ASTERIX structure
TYPE    = 'EXT     '           / Data type of this extension
PARENT  = 'MORE    '           / Parent HDU of this extension
CHILDREN=                    0 / Number of children for this HDU
END

XTENSION= 'IMAGE   '           / IMAGE extension
...
EXTNAME = 'HISTORY '           / Contains ASTERIX HISTORY structure
TYPE    = 'HISTORY '           / Data type of this extension
CREATED = '12-Feb-98 14:37:17' / Date & time of HISTORY creation
UPDATE  = 'NORMAL  '           / Update mode for further additions
PARENT  = 'PRIMARY '           / Parent HDU of this extension
CHILDREN=                    1 / Number of children for this HDU
CHILD1  = 'RECORD1 '           / Name of child extension
END

XTENSION= 'IMAGE   '           / IMAGE extension
```

```
...
EXTNAME = 'RECORD1 '           / Contains ASTERIX HIST_REC structure
TYPE    = 'HIST_REC'           / Data type of this extension
HOST    = 'xund    '           / Machine application was run from
DATE    = '12-Feb-98 14:37:17' / Date & time application was started
COMM    = 'WRITE1 V1.0'        / Version of application being run
TEXT_REC=                    0 / Current text line number
PARENT  = 'HISTORY '           / Parent HDU of this extension
CHILDREN=                    0 / Number of children for this HDU
END
```