

# Working with FITS data

# Outline

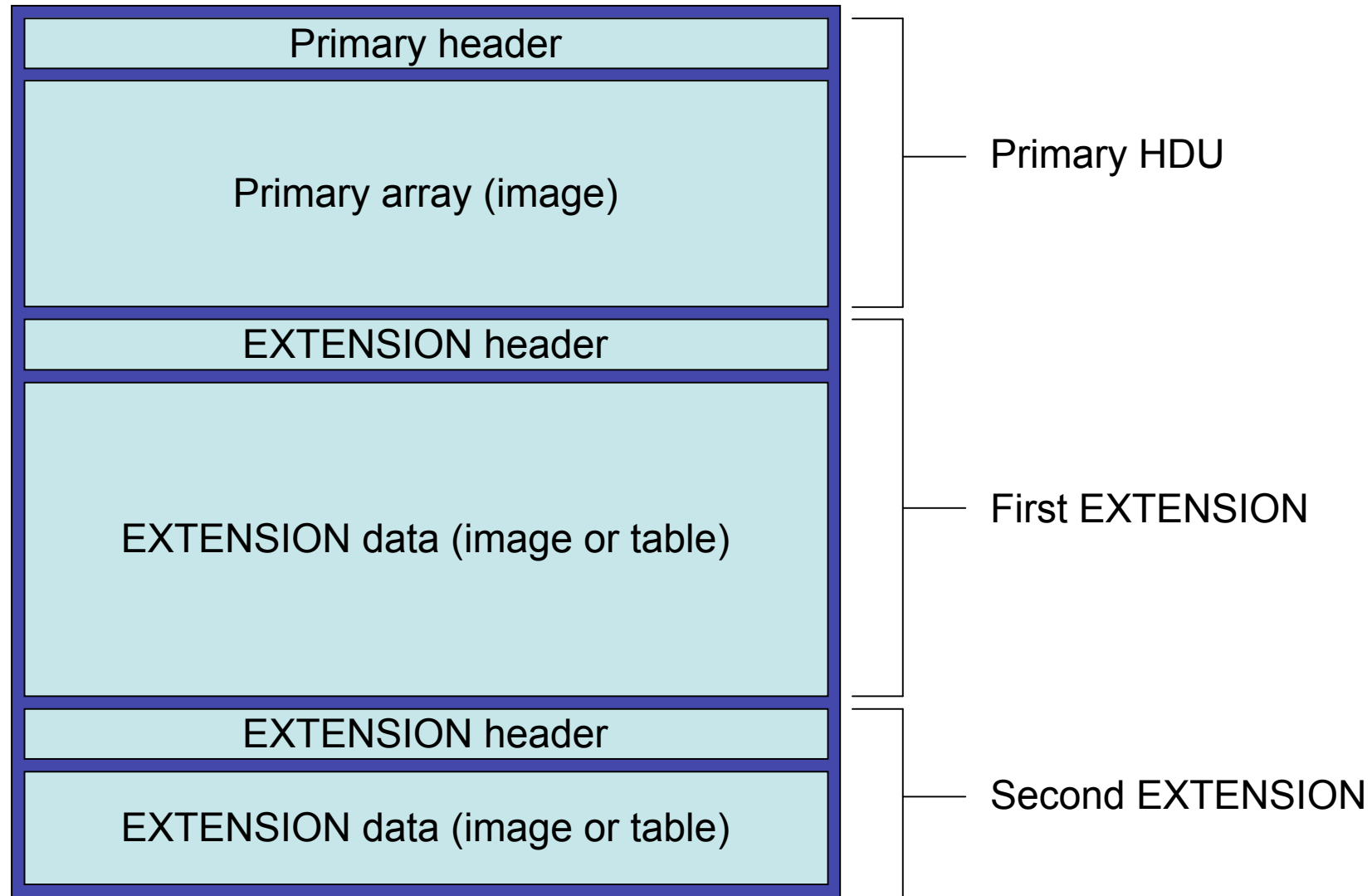
- What is a FITS file?
- Contents of a FITS file
  - headers
  - images
  - tables
- Software for manipulating FITS files
  - FTOOLS
  - FV, ds9, GAIA
- FITS for programmers
- Tricks & tips

# FITS and FTOOLS

- FITS = Flexible Image Transport System
  - It's a file format standard used extensively in high energy astronomy
  - It goes far beyond just images
  - Default format used by
    - XSPEC – spectral analysis
    - XIMAGE – image analysis
    - XRONOS – timing analysis
  - Used for ASCA, Einstein, CGRO, Rosat, XTE, and now INTEGRAL data

FITS info at <http://heasarc.gsfc.nasa.gov/docs/heasarc/fits.html>

# Anatomy of a FITS file



# Headers

- Conform to `KEYNAME = value / comment string`
- Example **primary** header:

```
SIMPLE = T           / file does conform to FITS standard
BITPIX = 16          / number of bits per data pixel
NAXIS  = 2           / number of data axes
NAXIS1 = 440         / length of data axis 1
NAXIS2 = 300         / length of data axis 2
```

**440x300 pixel image**

- Example **EXTENSION** header:

```
XTENSION= 'BINTABLE' / binary table extension
EXTNAME = 'MYDATA'   / data name
BITPIX  = 16         / number of bits per data pixel
NAXIS   = 2          / number of data axes
NAXIS1  = 440        / width of table in bytes
NAXIS2  = 300        / rows in table
```

•  
•

# Extensions

- Extensions are the main data storage area – the primary array is usually not used now.
- Images can contain IMAGES or TABLES
- Tables can be ASCII or binary
  - ASCII are faster to display
  - Binary are more compact
- Table columns can be vectors
  - n-dimensional structure
- Extensions are a good way to store several related but separate data structures in a single file, even if the data are of different types

# FTOOLS

- Provided/distributed by HEASARC
- Is a set of programs to display, edit, manipulate data in FITS files
- Standard interface (command-line driven)
- Also mission-specific tools (so are expecting data in a certain format) for use with ASCA, CGRO, ROSAT etc data

[http://heasarc.gsfc.nasa.gov/docs/software/ftools/ftools\\_menu.html](http://heasarc.gsfc.nasa.gov/docs/software/ftools/ftools_menu.html)

# First FTOOLS commands

- Starting FTOOLS
  - lhea-init shell script (may already be started)
- Getting help
  - **fhelp** <ftool> ... help pages for each FTOOL
  - **fapropos** <subject> ... search for FTOOLS which could help on a subject
  - **flaunch** ... a GUI launcher for all the FTOOLS and analysis packages (with built-in help)

# Using FTOOLS commands

- Each FTOOL has a set of parameters, which can be either mandatory or hidden
- See the parameters for an FTOOL with either fhhelp or plist
- Specify parameters and values on the command line or, if you don't...
  - Mandatory parameters will be requested
  - Hidden parameters will use defaults
- Specify hidden parameters on the command line to override the defaults. In fhhelp, hidden parameters are shown as (parameter)

# Specifying FITS filenames

- The filename can be used to specify file and/or extension in a variety of ways:
  - myfile.fits                      the file itself (depends on s/w how this is used)
  - myfile.fits[1]                    1st extension in myfile.fits
  - myfile.fits+1                    as above, but old-style notation
  - myfile.fits[EVENTS]              1st binary table with EXTNAME=EVENTS
- Prefix a filename with ! to let it be overwritten, or use the clobber parameter
- Use quotes or \ if you use [ ] or ! on a command line
  - fimgstat myfile.fits[4]            doesn't work
  - fimgstat myfile.fits¥[4]            does
- See later for rowfilter, colfilter and histogram options

# Top 10 FTOOLS



FV



fselect – filtering of event data in tables



fhisto/f2dhisto/fimghisto – histogramming



fstatistic/fimgstat – quick stats



farith – arithmetic on 2 images



fsort – sort a *small* table



fdiff – difference between 2 FITS files



fstruct – get structure of file

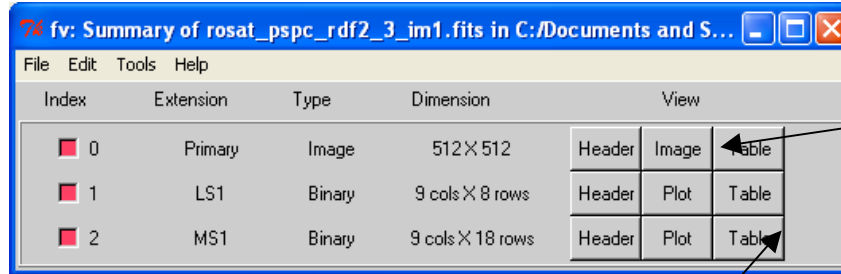
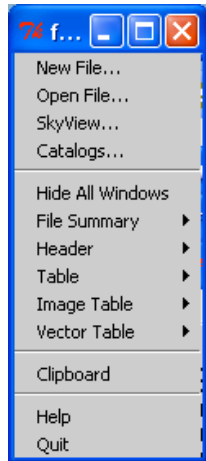


fdump – ASCII view of contents



fcreate – make a FITS file from scratch

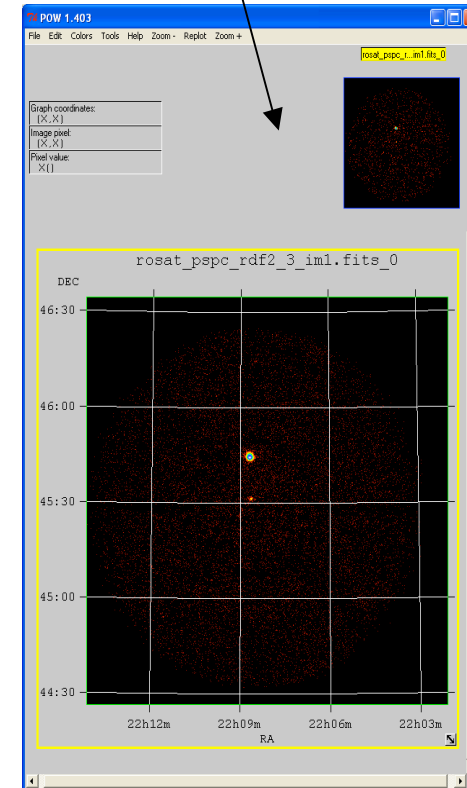
# FV – the #1 FTOOL



Click here for image display

Inspect table data

	SRC_NR	LEV1_X	LEV1_Y	NET_CTS	BKG_CTS	S_LIKE
	1I	1E	1E	1E	1E	1E
	NONE	pixel	pixel	count /cell	count /pixel	NONE
1	1	2.305500E+002	1.982200E+002	6.266500E+002	1.400000E-001	4.710000E+003
2	2	2.332100E+002	1.989500E+002	1.489460E+003	1.400000E-001	1.122400E+004
3	2	2.282500E+002	1.997300E+002	5.458700E+002	1.400000E-001	4.091000E+003
4	4	2.335500E+002	2.015000E+002	3.763800E+003	1.400000E-001	2.831400E+004
5	5	2.281300E+002	2.023900E+002	1.045850E+003	1.400000E-001	7.827000E+003
6	6	2.305500E+002	2.042200E+002	3.461240E+003	1.400000E-001	2.602300E+004
7	7	2.333100E+002	2.040600E+002	6.665090E+003	1.400000E-001	3.315000E+003
8	8	2.815000E+002	2.135000E+002	7.350000E+000	1.400000E-001	8.000000E+000
9	9	2.275000E+002	2.155000E+002	7.350000E+000	1.400000E-001	8.000000E+000
10	2	2.333400E+002	2.541800E+002	9.197000E+001	1.400000E-001	2.310000E+002
11	11	2.377900E+002	2.545000E+002	7.330000E+000	1.400000E-001	8.000000E+000
12	12	2.210000E+002	2.685000E+002	7.420000E+000	1.300000E-001	9.000000E+000
13	13	2.765000E+002	2.745000E+002	7.390000E+000	1.400000E-001	8.000000E+000
14	14	2.162600E+002	2.762100E+002	7.460000E+000	1.300000E-001	9.000000E+000
15	15	3.050600E+002	2.875000E+002	7.680000E+000	1.100000E-001	9.000000E+000
16	16	3.914000E+002	3.846900E+002	1.811000E+001	8.000000E-002	8.000000E+000
17	17	1.100000E+002	3.980000E+002	1.911000E+001	6.000000E-002	1.100000E+001
18	18	2.440000E+002	7.949000E+001	3.333000E+001	6.000000E-002	1.100000E+001



# Alternatives to FTOOLS

- Other visualisation tools exist
  - SAOimage
  - DS9
  - Gaia
  - XIMAGE
- The DIY approach (programming)

# FITS programming

- For when the standard TOOLS don't do what you want...

Language	For FITS
FORTRAN	fitsio
C	cfitsio, DAL
C++	cfitsio, CCfitsio
Perl	Perl/cfitsio
IDL	astrolib

- Watch out for 'historical' FORTRAN-style array ordering !

<http://heasarc.gsfc.nasa.gov/docs/software/fitsio/fitsio.html>

# Advanced FITSing (1)

- Assuming we have an events list in first extension of myfile.fits...
- Row filtering on the command line
  - `myfile.fits[1][QUALITY>5]`
    - select only the rows in which the QUALITY column has a value > 5
  - `myfile.fits[1] [#row >= 125 && #row <= 175]`
    - select rows 125-175
- Histogramming on the command line
  - `myfile.fits[1] [bini detx, dety]`
    - Creates a temporary 2D histogram of integer values in detx and dety columns. Use default limits and bin size
  - `myfile.fits[1][bin time=TSTART:TSTOP:0.1]`
    - creates a histogram of values in time column, limits are taken from the TSTART and TSTOP keywords in header, bin size is 0.1

# Advanced FITSing (2)

- **GROUPING** and hierarchical data
  - Grouping uses special tables with EXTNAME = GROUPING which contain links to data structures, or other GROUPING structures.
  - A large data structure can be split over several physical files.
  - Software (cfitsio or DAL) can then search the data tree for a specific structure without the user needing to know where it is.