LECTURE 6

Scientific Data Formats

Data formats are:

- Critical to science
- Often overlooked

Criteria for data formats are:

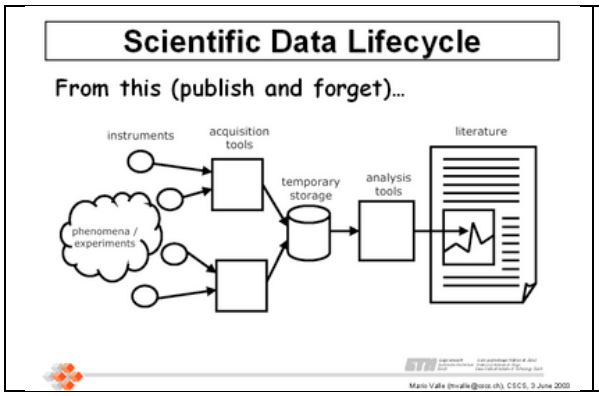
- Easy to use the data
- Preserves the original information
- Allows archiving
- Allows use by future researchers

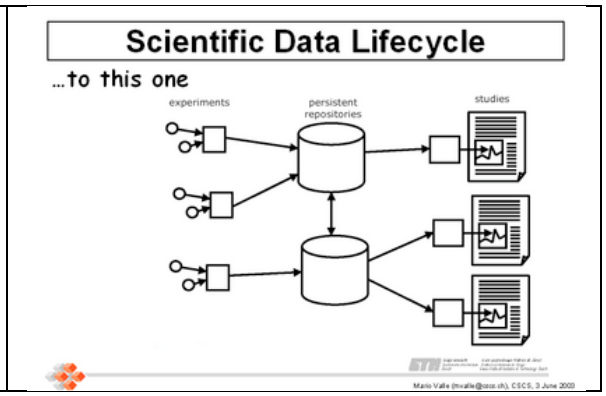
Goals for data formats:

1. Store data in a way that is
    a. Convenient
    b. Flexible
    c. Portable
2. Facilitate long-term archiving
    a. Preserve data
    b. Easy to search and retrieve
    c. Easy to use

The Scientific Data Lifecycle

Then… and Now…





Then…

- Data was published in the literature
- Advantages
  - Easy to find
  - Data with the results
- Disadvantages
  - Data in printed form
  - Hard to reuse data
  - Large data sets generally not published

Now…

- Data is archived
- Advantages
  - Data can be searched
  - Data is in electronic (usable) form
  - Very large data sets can be archived

Data Format Requirements

- Portability
  - Need to be able to share data
  - Proprietary formats (e.g. GIF)
- Lossless
  - No loss of information in storage or compression
  - High level data products can be truncated at the precision of the data
- Indexable
  - Need to know what the data is
  - Searchability
  - Self-describing data formats

ASCII

- American Standard Code for Information Exchange
- De facto standard for text on computers

Example ASCII Data File

```
# Offsets of short-hard GRBs from their putative host galaxies.
#
# Sources: 1 - Berger et al. (2005, Nature, 438, 988)
#          2 - Fox et al. (2005, Nature, 437, 845)
#          3 - Soderberg et al. (2006, astro-ph/0601455)
#          4 - astro-ph/0901.4038v2
# GRB     kpc    r/r_e   Source
#
 050509   44      13      2
 050709    3.8     1.8    2
 050724    2.57    0.4    1
 051221    0.760   0.29   3
```

File Contains

- Meta data—self-describing
- Four data columns

Advantages of Plain Text Data

- Easy to create files
- Good for quick-and-dirty data storage
- Easy to edit
- Good for intermediate data products
    - Plotting
    - Quick statistical calculations
    - Isolating subsets of data
    - “Playing” with data

Disadvantages of Plain Text Data

```
# Offsets of short-hard GRBs from their putative host galaxies.
#
# Sources: 1 - Berger et al. (2005, Nature, 438, 988)
#          2 - Fox et al. (2005, Nature, 437, 845)
#          3 - Soderberg et al. (2006, astro-ph/0601455)
#          4 - astro-ph/0901.4038v2
# GRB    kpc    r/r_e   Source
#
 050509  44     13      2
 050709   3.8    1.8    2
 050724   2.57   0.4    1
 051221   0.760  0.29   3
```

- Not inherently self-documenting
  - What does “r/r_e” mean?
  - What physical quantity is “kpc”?
- File structure is not specified
  - Is this a matrix, or four columns of data, or something else?

```
1.012   2.111  2.111  0.000
2.909   8.321  4.534  0.000
5.132   9.876  1.357  0.000
0.000   0.000  0.000  0.000
```

- Not efficient for large datasets, such as images.

XML

- Extensible Mark-Up Language
- System for making plain text files self-documenting

Example of XML Mark-Up

```
<photometry>
 <photometry source="IMC930475">
  <date>2008-04-14</date>
  <starttime>03:12:15</starttime>
  <exposure>2000</exposure>
  <filter>R</filter>
  <magnitude>15.87</magnitude>
  <magerror>0.03</magerror>
 </source>
 <photometry source="IMC930476">
  <date>2007-02-28</date>
  <starttime>11:42:04</starttime>
  <exposure>2000</exposure>
  <filter>R</filter>
  <magnitude>18.23</magnitude>
  <magerror>0.15</magerror>
 </source>
</photometry>
```

- Example contains
    - Photometry data
    - Description of what the data is

Advantages of XML

- Files are self-documenting
- Structure is self-documenting

Disadvantages of XMK

- Files can be very large
- Not always suitable for very large data sets
- Not easy to read by a human

FITS

- Flexible Image Transport System
- Developed in 1970s to store and transport astronomical images
- Grown into a general data storage format

Format of a FITS File

- Files segmented into "Header + Data Units" (HDUs)
- 1st (primary) HDU contains
    - Keywords describing the data, and
        - *N*-dimensional data set in binary form, or
        - No data
- Remaining HDUs (called extensions) contain
    - Keywords
    - Data

Three types of data

- Image
  An *N*-dimensional array of data in binary format
- ASCII Table
  A table (rows and columns) of data stored in ASCII format
- Binary Table
  A table (rows and columns) of data stored in binary format

Keywords

- Meta data that describes the data, or contains additional information that is related to the data.

  KEYNAME = value / comment string

- Strict rules on keyword names and values
  - Can only used approved characters
  - Can only be approved data types (e.g., integer, floating point, etc.)

Required keywords

```
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
```

Describe basic structure of the HDU

Example of FITS Format File

- ftlist example.fits H

```
Name      Type                Dimensions
----      ----                ----------
HDU 1     Primary Array       Null Array
HDU 2     wh250005527I        Image          Real4(1138x1141)
HDU 3     MAGHIST             BinTable       69 cols x 5 rows
```

Can mix different types of data (Image and Binary Table) in one file.

HDUs are independent of each other

FITS Software

- Humans can not directly read or modify FITS files.
- Standard software package is FTools
- Standard programming library is FITSIO
- FITSIO has been ported to many languages

- C++
- C#
- Perl
- Tcl
- Python
- Ruby
- S-lang
- MatLab
- LabVIEW
- Photoshop

ftlist

Intended to print the contents of a FITS file in a flexible way.

See the online help file for details.

Five modes

- H—prints a summary of the files structure
- C—prints a list of the column names
- K—prints the keywords
- I—prints pixel values from an image
- T—prints data from a table

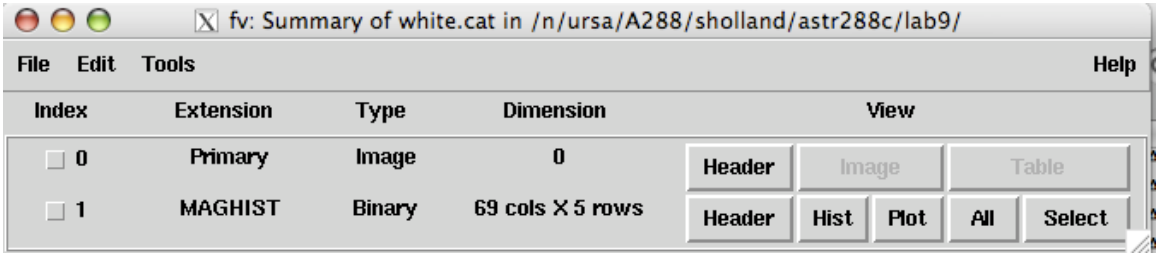
Ftlist is a *very* useful tool

fv

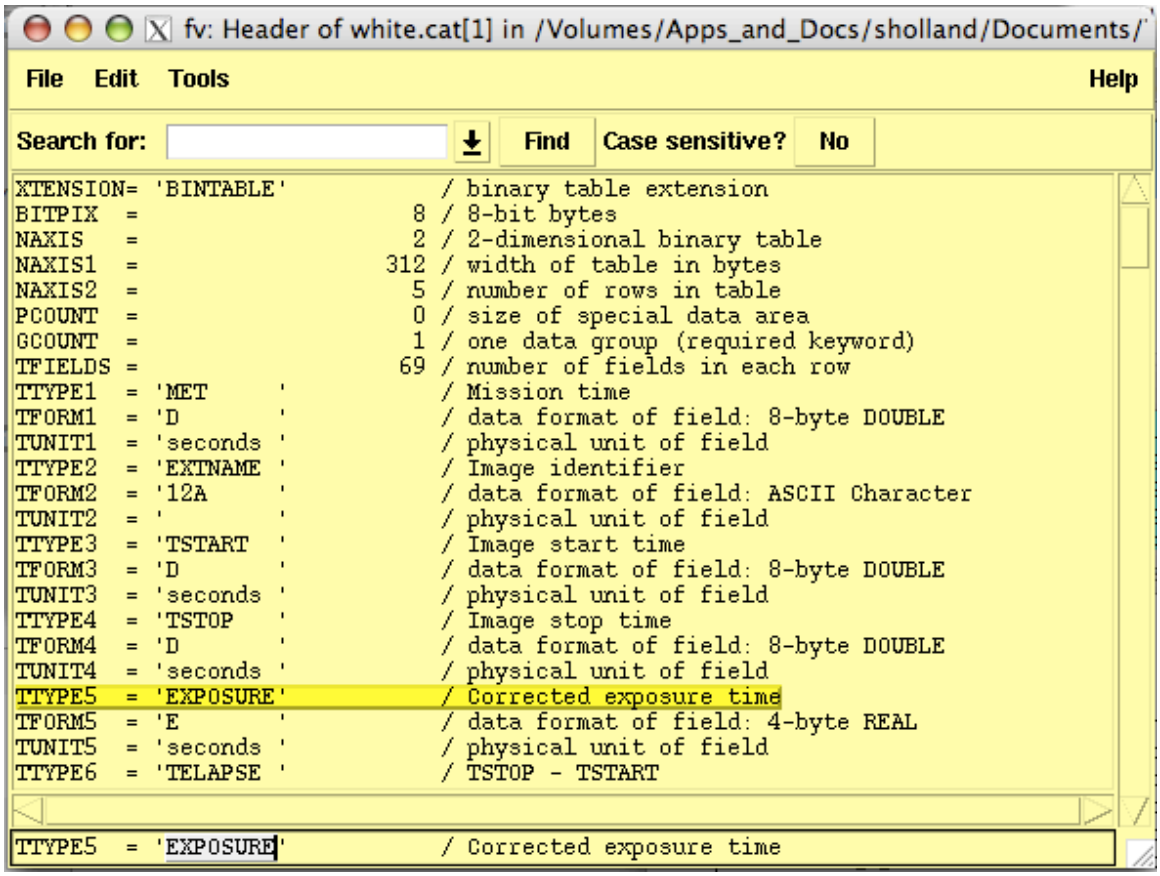
Interactive browser for FITS files.

Allows the user to view and modify the contents of a FITS file, and to create FITS files.

An open FITS files looks like this in FV



The keyword viewer looks like this

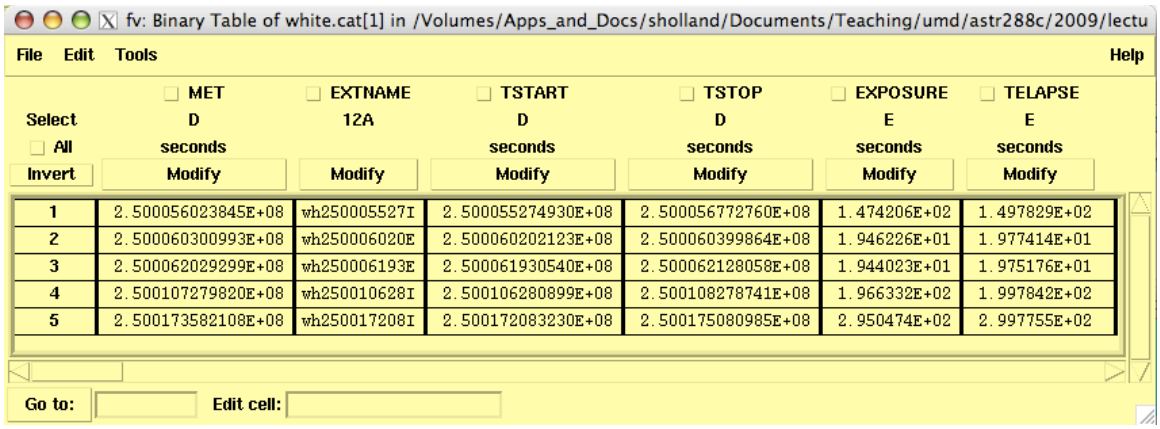


Edit a keyword by selecting it. The editing is done in the box at the bottom of the window. *Must press return in this box for an edit to take effect.*

Adding and deleting keywords is done from the "Edit" menu.

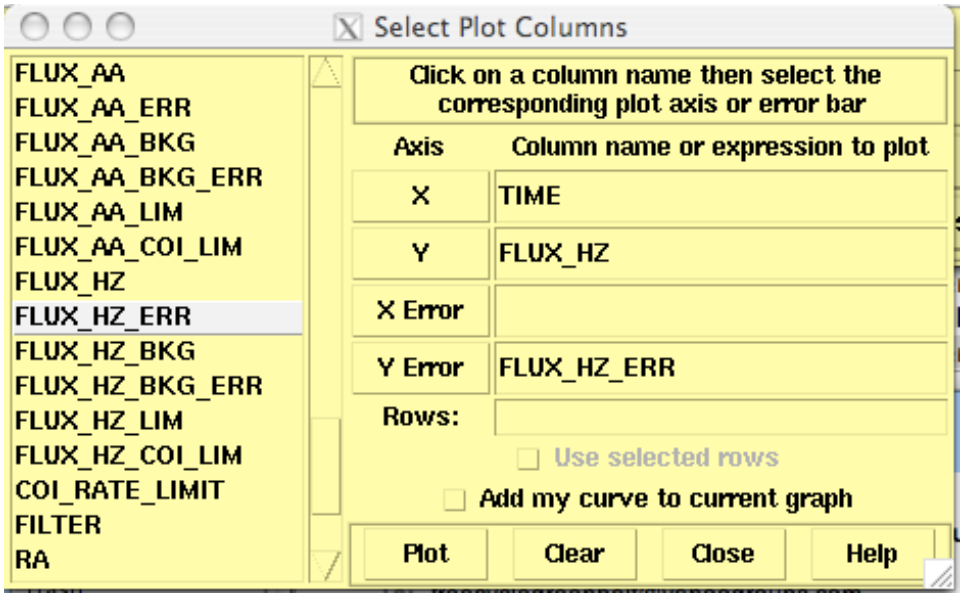
Be sure to save changes using the "File" menu.

The table viewer looks like this



Editing and saving is done the same way as for the keyword viewer.

The plotting tool looks like this



Select the columns to plot along the *X* and *Y* axes. Error bars can be set for each axis.