

# AGA5802

## Fits files

Prof. Alessandro Ederoclite

# A bit of history...

When digital astronomy was born, different communities embarked in different endeavours. The main ones were:

- US-based:
  - IRAF - Image Reduction and Analysis Facility (the only survivor)
- ESO-based:
  - MIDAS - <https://www.eso.org/sci/software/esomidas//>
- UK-based:
  - Starlink (when this word could still be used) <http://starlink.eao.hawaii.edu/starlink>

All different file types and utilities.

# Fits...

The “fits” format is born in that framework. It is relatively close to MIDAS’s “mt” format and with a nice separation between header and image as IRAF’s “imh/pix”

“Fits” stays for “Flexible Image Transfer System

This format is endorsed by the International Astronomical Union; here is the webpage of the fits working group <https://fits.gsfc.nasa.gov/>

Other formats (e.g. theHDF5) are used in astronomy (*but we will focus on fits*).

# Fits - the structure

In general, a fits has two parts:

- The “header”, this is so important that we will discuss it in some detail.
- The data; the data... well, is the data.

In some cases, you can stick together different fits files; each with its header and data section. This is what we call a “multi-extension” fits file (more on this later).

# Let's give a look at an image

Let's open the image: f606\_mosaic\_v2.fits

But let's be crazy! Do:

```
less f606_mosaic_v2.fits
```

Is this a text file?

```
SIMPLE = T / FITS STANDARD
BITPIX = -32 / FITS BITS/PIXEL
NAXIS = 2 / NUMBER OF AXES
NAXIS1 = 4096 /
NAXIS2 = 4096 /
OBJECT = 'F606_MOSAIC_V2[1/1]' /
ORIGIN = 'KPNO-IRAF' /
DATE = '30-10-96' /
IRAFNAME= 'f606_mosaic_v2.hhh' / NAME OF IRAF IMAGE FILE
IRAF-MAX= 0.000000E0 / DATA MAX
IRAF-MIN= 0.000000E0 / DATA MIN
IRAF-BPX= 32 / DATA BITS/PIXEL
IRAFTYPE= 'REAL' / PIXEL TYPE
DATAMIN = 0.
DATAMAX = 0.
CRPIX1 = 3008.236
CRVAL1 = 189.18661365727
CTYPE1 = 'RA---TAN'
CD1_1 = -1.013372E-5
CD2_1 = -4.466669E-6
CRPIX2 = 1166.667
CRVAL2 = 62.2036751382
CTYPE2 = 'DEC--TAN'
f606_mosaic_v2.fits
```

Clearly something happens at that “END”  
Because things after the “END” get CRAZY

```
D15MASK = 'f606_1c_v2.hhh' / Drizzle, input data quality mask image
D15WTSCL= 1. / Drizzle, weighting factor for input image
D15OUTDA= 'f606_mosaic_v2.hhh' / Drizzle, output data image
D15OUTWE= 'f606_mosaicc_v2.hhh' / Drizzle, output weighting image
D15COEFF= ' ' / Drizzle, coefficients file name
D15SHFTU= 'input ' / Drizzle, units used for shifts (output or input)
D15PIXFR= 1. / Drizzle, linear size of drop
D15LAM = 0. / Drizzle, wavelength applied for transformation
D15SCALE= 1. / Drizzle, scale (pixel size) of output image
D15ROT = 90.50999 / Drizzle, rotation angle, degrees anticlockwise
D15XSH = -450.329 / Drizzle, X shift applied
D15YSH = 417.38 / Drizzle, Y shift applied
D15OUTNX= 4096 / Drizzle, X size of output image (if newly creat
D15OUTNY= 4096 / Drizzle, Y size of output image (if newly creat
D15EXPSC= F / Drizzle, were images converted to counts-per-se
D15FILVA= 'INDEF ' / Drizzle, fill value for zero weight output pixe
END
```



# The “END”, my friend

“END” is the keyword which tells the fits standard where the header ends.

It is a reserved keyword and its use is very very special.

# Data in a fits

Fits can host two types of data:

- “Images” - The easiest way to imagine the content of a fits file is an image. The interesting thing that an “image” can have one dimension (a spectrum), two dimensions (an image as you would expect it) or three (a datacube; we will come back to this type of data when speaking about spectroscopy)
- “Tables” - As you would expect, this is a table, with many columns and rows. *Curiosity*: the SDSS spectra are stored as fits tables.



# The header

The header carries the information about your data: the “metadata” (*fashion!*).

If you have a collection of files, the first thing you (should) always do is to check what they are. You have two options: either check every object one by one (good luck in telling if you are looking at an image in the *r* filter or the *g* filter) or check what the header tells you.

Note: if the header is not properly filled, you are in **big** troubles.

# The “keywords”

The header is structured in terms of “keywords” which carry some information.

Example:

```
OPSIZE = 512 / PSIZE of original image
```

The first 8 characters are the name of the keyword, after the “=” sign it’s the content of the keyword and after the “/” an explanation (sometimes needed)

ESO found a loophole in the standard with the (in)famous “HIERARCH” keywords.

Example:

```
HIERARCH OAJ PRO FWHMSEXT = 0.87 / FWHM arcsec estimated  
with SE
```

# Important fits keywords

The keywords you are normally interested tell you:

- What (you are observing)
  - OBJECT
- Where (your telescope is pointing)
  - RA
  - DEC
- When (you are observing)
  - DATE-OBS
  - EXPTIME
- How (you are observing)
  - FILTER, GRATING,...

# Important fits keywords (contd.)

These keywords are not standard and different observatories can have different standards (within an observatory, people tend to keep the same standards... “tend to”).

Yet, even with variations, most of these keywords are always present.

# “Technical” header keywords

Some keywords are worth knowing:

- SIMPLE; if “T” (“true”) it means it is a fits file.
- BITPIX; it’s the bits per pixel, most used are 16 or 32 (signed or not)
- NAXIS is the number of axis: 1 is for spectra, 2 for images, 3 for datacubes...
- OBSERVER and INSTRUME are often present (ESO’s “I, CONDOR” as observer at La Silla is legendary)

# The World Coordinate System

You hardly ever mess around with the WCS yourself and it is often nicely provided by the observatory.

How do we go from “pixel coordinates” to RA,Dec?

In fact, there are two types of “pixel coordinates”:

- Physical coordinates: the coordinates in terms of pixels which physically exist
- Logical coordinates: the coordinates taking into account pixels which do not physically exist (like a pre-scan region, which may or may not be present)

# World Coordinate System (contd.)

At first order, what the WCS does is to give a reference, a rotation and a scale to the image.

The reserved keywords are:

CRPIX $i$  - the reference pixel in the  $i$ -th coordinate (x, y, ....)

CRVAL $i$  - the value of the reference pixel

CTYPE $i$  - the type of projection used

CD $i_j$  - this is a rotation matrix; note that the scale of the pixel is embedded here

Note the use of  $i$  and  $j$  and not, necessarily 1 and 2; i.e. this can be anything

# A Multi-Extension Fits File (MEF)

In a MEF, the limit is your imagination. A couple of good examples:

If you have multiple CCDs on your detector, you will have:

- Extension 0: it is a “global header” with general information about the observation
- Extension  $i$ : image taken with your  $i$ -th CCD with its header and image

Another example can be:

- Extension 0: the “global header”
- Extension 1: an image
- Extension 2: a table with the objects detected on the image



# Checking the image!

Let's look at the bloody image, shall we?

You can either open it like:

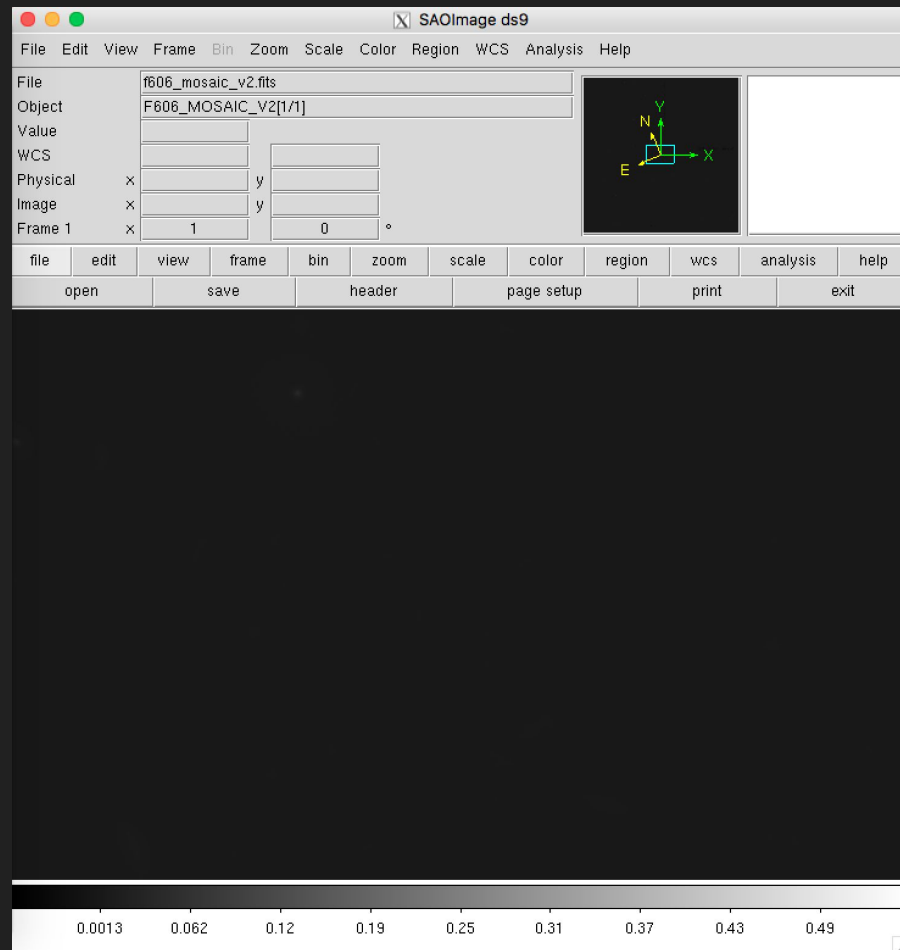
```
ds9 f606_mosaic_v2.fits
```

Or

```
ds9
```

Then go to “File” -> “Open”

**An awesome black image!**



# Check ds9

There is the “usual” menu.

The coordinates of your cursor (in pixels and “CCD coordinates”)

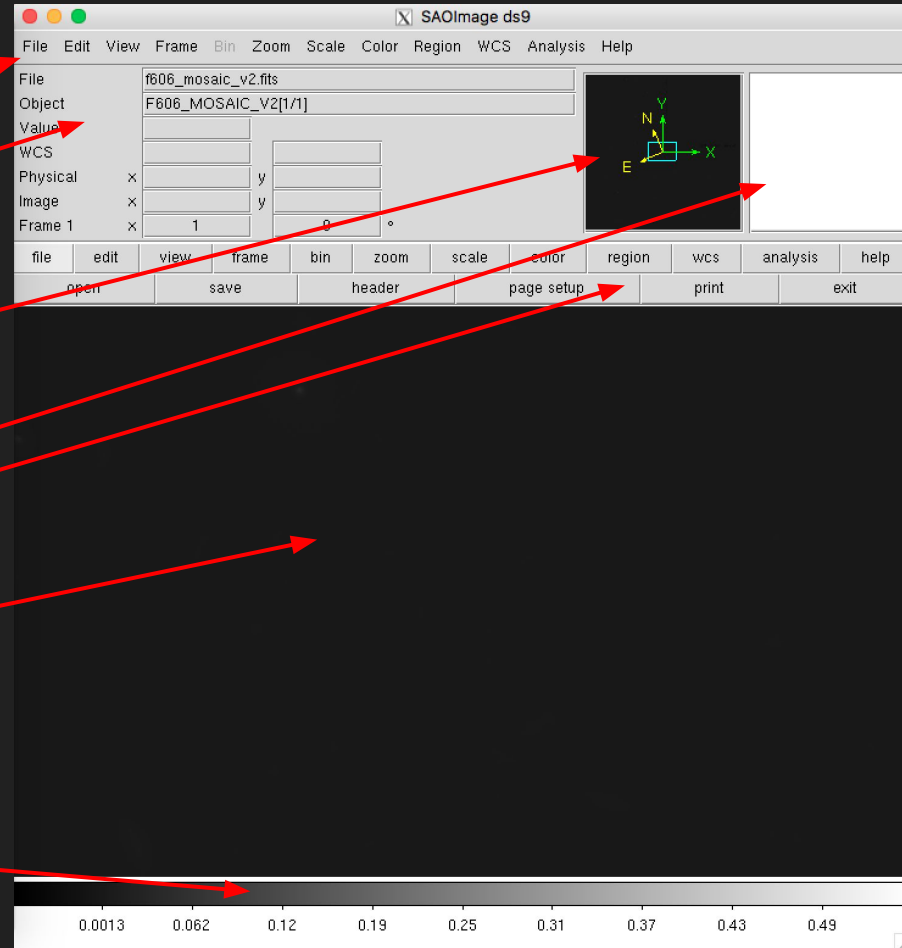
A global view of the image (with orientations!)

A zoom on the position of your cursor

A series of shortcuts

A view of your image

The colour bar



# Change the contrast

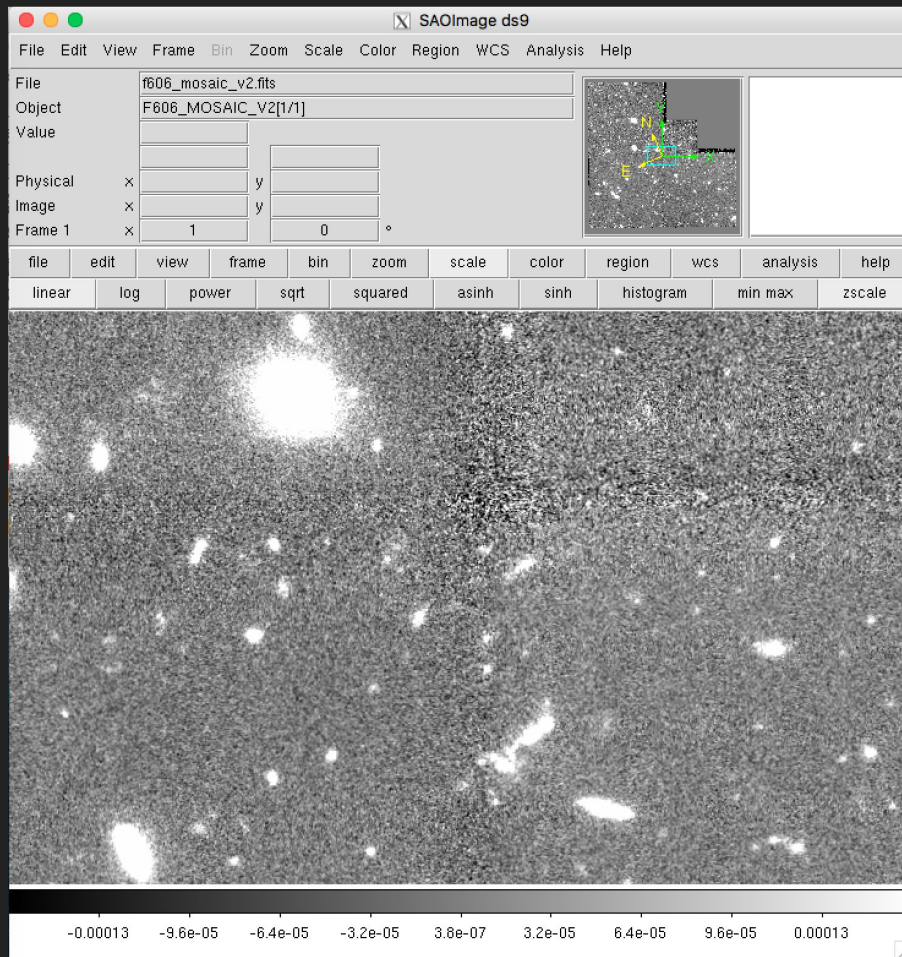
You can play with the scale of the image in many ways.

What I like as a “first step” is:

“Scale” -> “zscale”

**Magic** :-)

*I encourage you to play around with the other parameters*

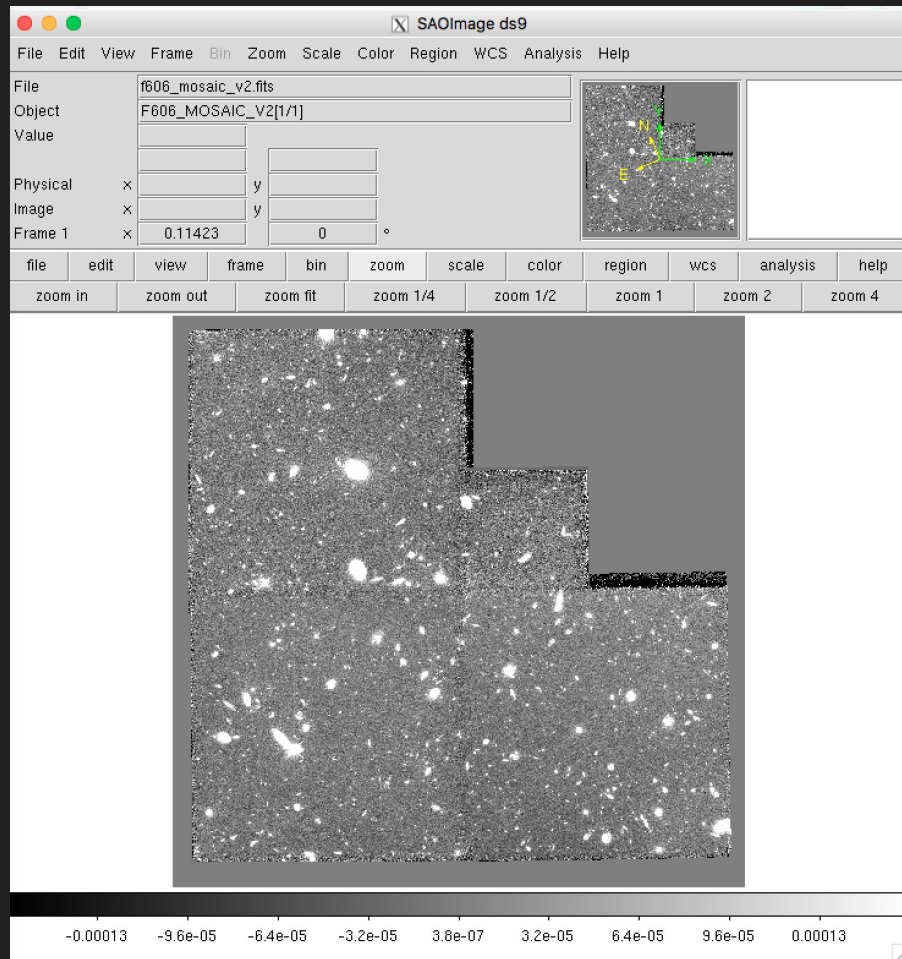


# Zoom

If you want to see the full image, you should go to:

“Zoom” -> “Zoom fit”

You will see the characteristic shape of the WFPC2 mosaic.



# Exercise

Do you have any fits images? Open them with ds9

If you don't, check the manual.

Check how to measure the background level of an image.

Check how to get the “histogram” of an image.