University of Glasgow

SSW, Radio, X-ray, and data analysis

Eduard Kontar

School of Physics and Astronomy
University of Glasgow, UK

Pre-school installation guide for IDL, SSW and Perl ([here]())

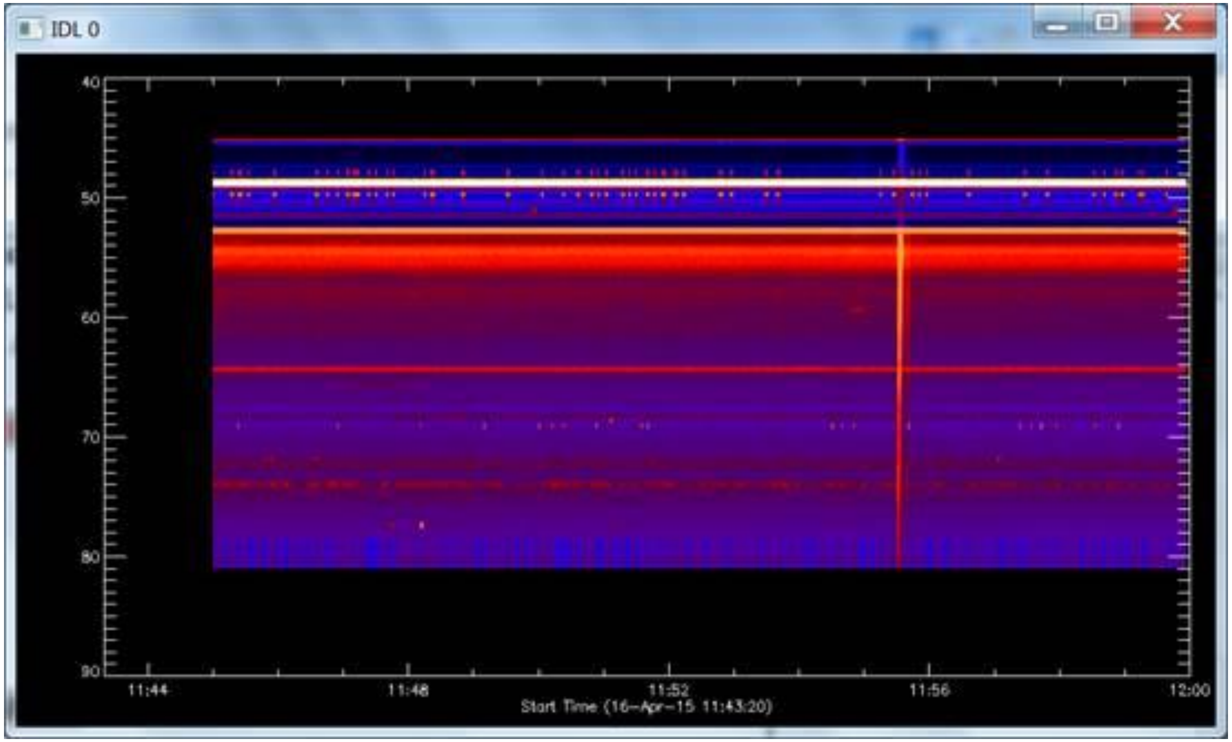
To upgrade SSW (requires Perl installation) (extensive FAQ [here]())

IDL> **ssw_upgrade**,/loud,/spawn,/passive_ftp
To check your installation open RHESSI GUI
IDL> **hessi**
To add new instrument (e.g. Nancay which is a part of **radio** package):
IDL> **setssw_windows**, /nrh
* Following instruments selected:
radio/nrh
* Mission level setup files
* Instrument level setup files
* Personal startup files
Executing: C:\ssw\site\setup\PERSONAL_STARTUP_WINDOWS.pro

Plotting dynamic radio spectrum

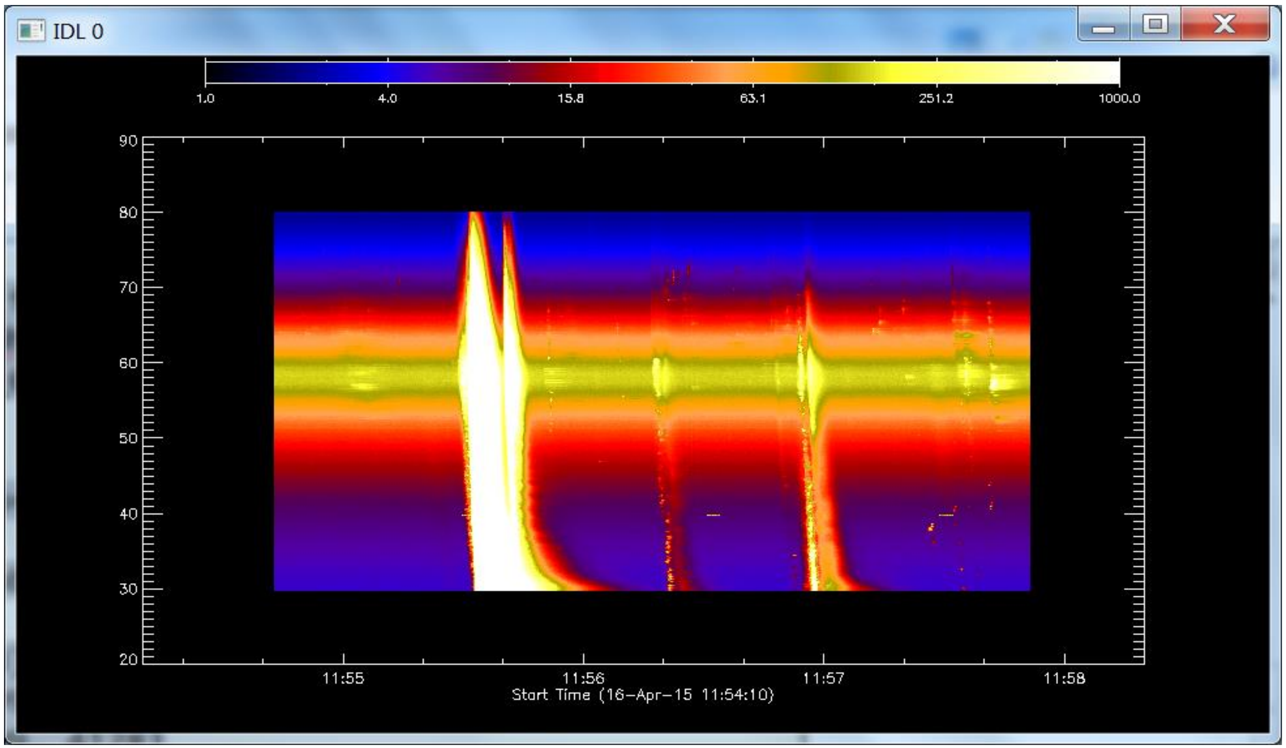
To analyse and plot the radio spectrum, use IDL based solar software SSW with 'radio' package required (documentation is available here). As an example, one can use this simple code

```
IDL> radio_spectro_fits_read,'GLASGOW_20150416_114500_59.fit',data,time,freq
IDL> loadct,5
% LOADCT: Loading table STD GAMMA-II
IDL> spectro_plot,data,time,freq
% Compiled module: SPECTRO_PLOT.
```



```
IDL> trange='2015-Apr-16 '+['11:55:20','11:56:00']
IDL> spectro_plot2,data,time,freq,timerange=trange,
yrange=[50,80]

IDL> data_bg = constbacksub(data,/auto)
% Compiled module: CONSTBACKSUB.
Running Constant Backsub algorithm Version 2.1
```

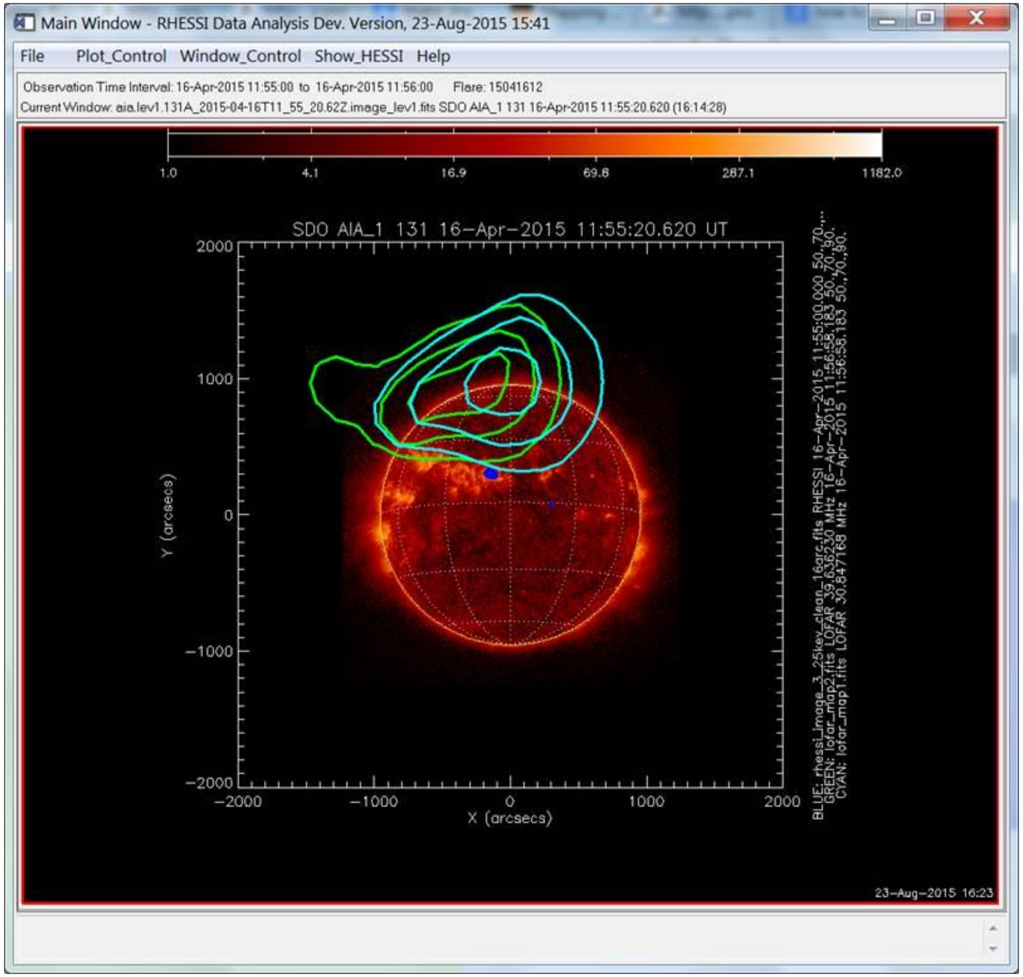


Problem: Estimate Brightness Temperature

Find the brightness temperature from e.g. LOFAR images at ~40 MHz?

$$I_\nu = \frac{2\nu^2 k_b T}{c^2}$$

For info: the intensity at low frequencies can be written



Problem: thermal energy flare

Find the thermal energy of a solar flare (e.g. 06-Jan-04 7:30)?

For info:

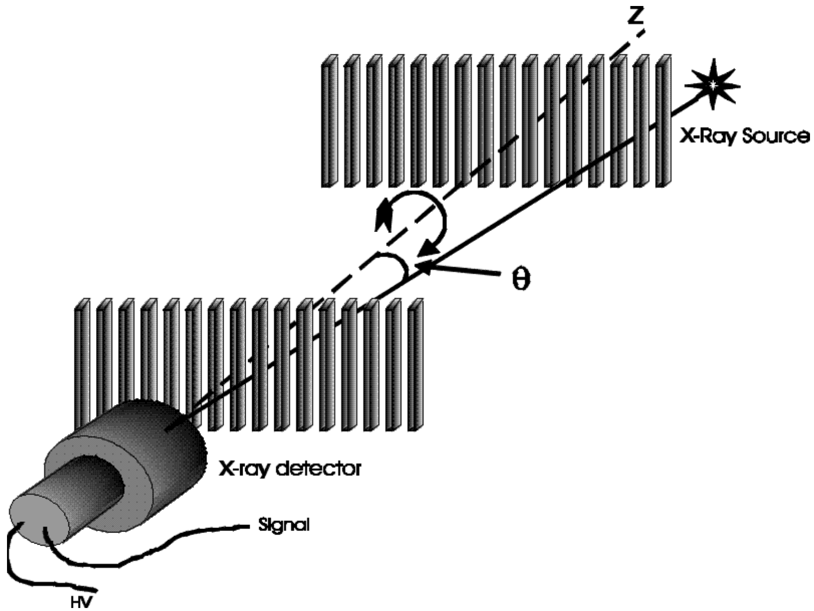
Energy = 3*N*kB*T

where N=n*V, n – number density, V – volume

Using the observable parameters:

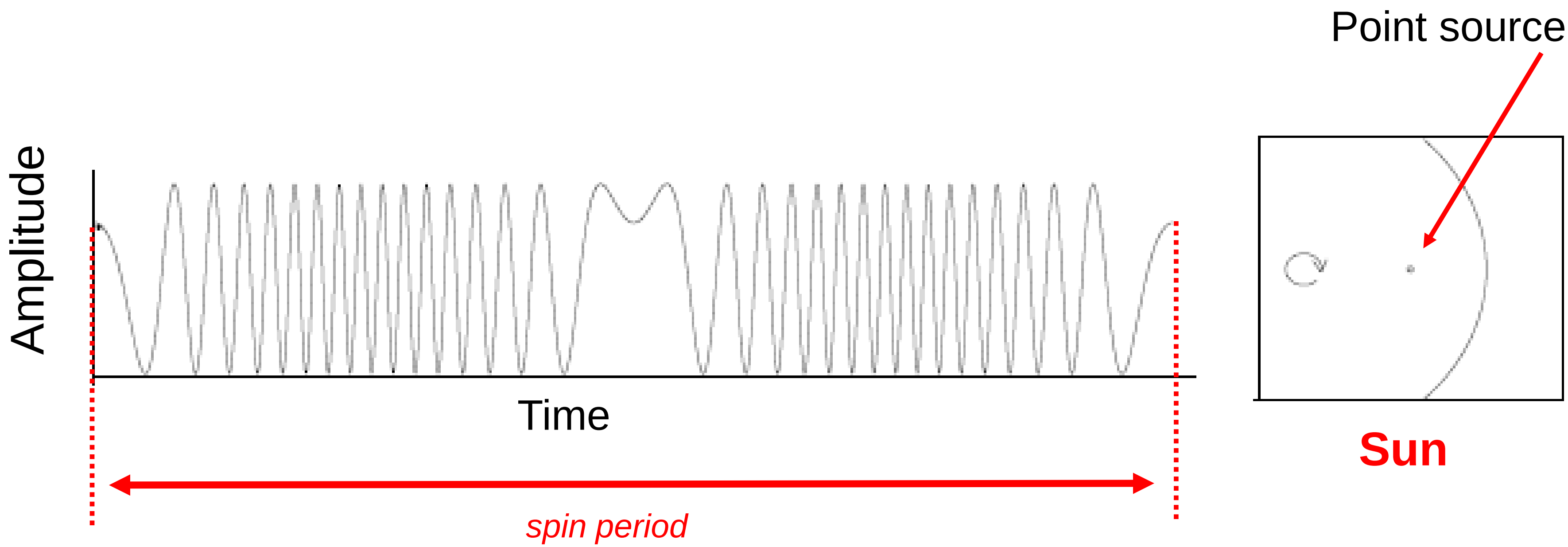
$EM=n^2V$

Rotating Modulating Collimators



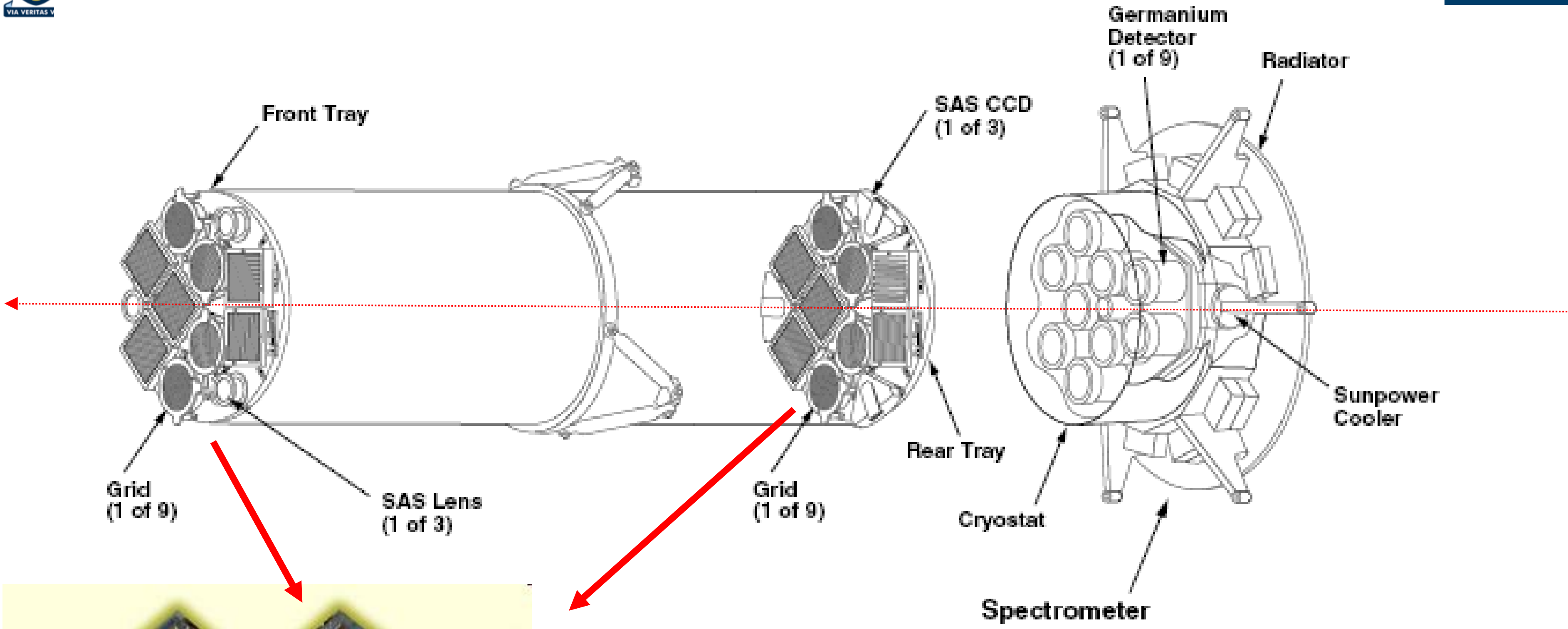
RHESSI detectors look at the source through a pair of grids called **Rotating Modulating Collimator** (RMC)

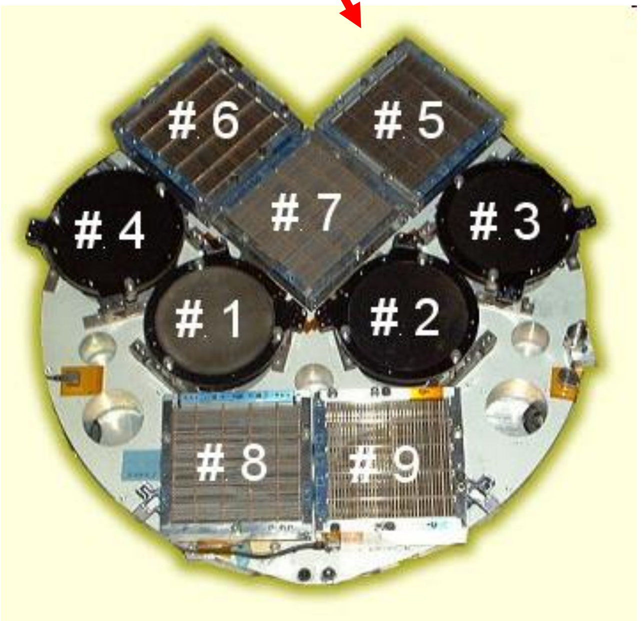
Spacecraft spins about once every ~4 sec => ***artificial modulation of incoming X-ray flux***



RHESSI imaging

SUN

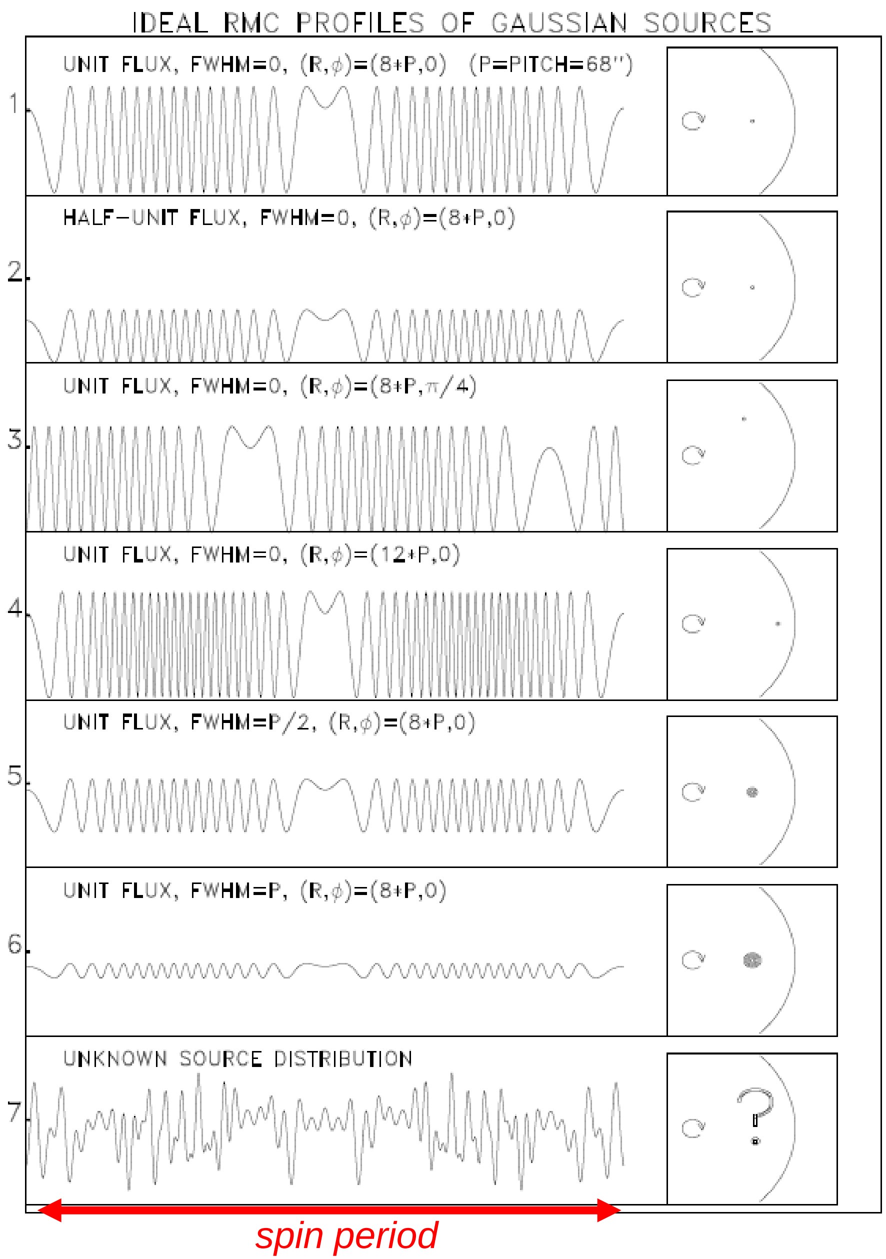




RHESSI has 9 RMCs for 9 detectors

Slats/Slits spacing growing with detector (RMC) number
⇒ angular resolution from ~2.3'' (RMC #1) to 180'' (RMC #9)

RHESSI: ideal modulated lightcurves



Modulation profiles for various ideal sources for a grid of pitch ***P*** with equal slits and slats

Point source

Half flux from the point source => note half amplitude

45 degrees angle => note change of phase

Source further from the axis => note change of modulation frequency

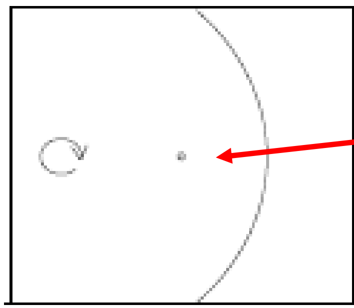
Source size=P/2 => note change of the amplitude

Source size=P => note change of modulation depth (no modulation for source size >> P)

Modulation encodes spatial source information:
Phase of the modulation => position angle
Distance from the centre => modulation frequency
Amplitude => source size

Modulated lightcurves

Sun



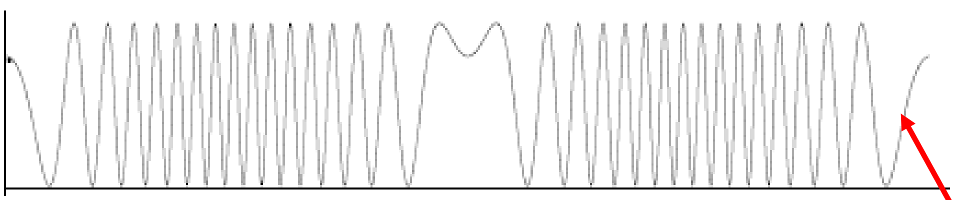
Point source

Incoming **photon flux** from ***pixel m*** of the source

Time interval = observation time – dead time

Detector area

$$C_i = A \sum_m P_{im} F_m \Delta t_i.$$



Modulated Lightcurve

Probability to find a photon from ***m-th pixel*** in ***i-th time bin (basically the response of the instrument)***

To find an image is to find the solution:

$$I_m = \frac{1}{A} \sum_{i=1}^{N} [C_i / \Delta t_i] \widehat{P}_{im},$$

Photons (counts) in i-th time bin

RHESSI imaging

The fundamental problem of RHESSI imaging is to find the spatial photon distribution knowing **the modulated time profile** or ***visibilities*** (**solve an inverse problem! ;(**):

$$I_m = \frac{1}{A}\sum_{i=1}^{N}[C_i/\Delta t_i]\widehat{P}_{im},$$

To accomplish this task various imaging algorithms to solve this inverse problem exist:

Back Projection
CLEAN
Maximum Entropy Method MEM based (e.g. MEM NJIT)
PIXON
Forward Fit
Interpolated (smooth) FFT

…

You method could be here!

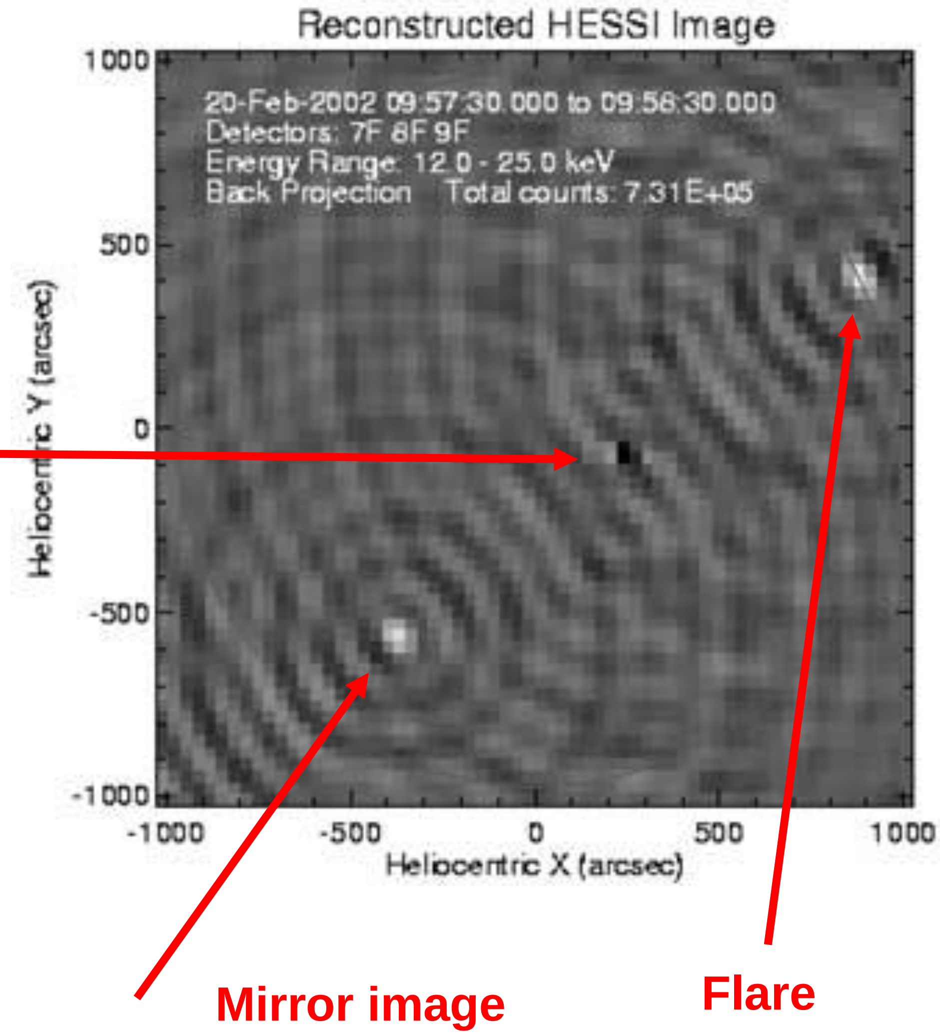
University of Glasgow

RHESSI imaging: Back Projection

Back projection (Mertz, Nakano, and Kilner, 1986) is the most basic method of image reconstruction (roughly 2D Fourier transforms(Kilner and Nakano, 1989)), leads to so-called ‘dirty map’ or ‘dirty image’.

$$I_m = \frac{1}{A}\sum_{i=1}^{N}[\mathcal{C}_i/\Delta t_i]\widehat{P}_{im},$$

$$\widetilde{P}_{im} = P_{im} - \langle P_m \rangle \qquad \widehat{P}_{im} = \widetilde{P}_{im}/\langle \widetilde{P}_m^2 \rangle.$$



Advantages:

- Very fast
- Linear and simple

Disadvantages:

- poor quality images
- no reliable source sizes

RHESSI imaging: CLEAN

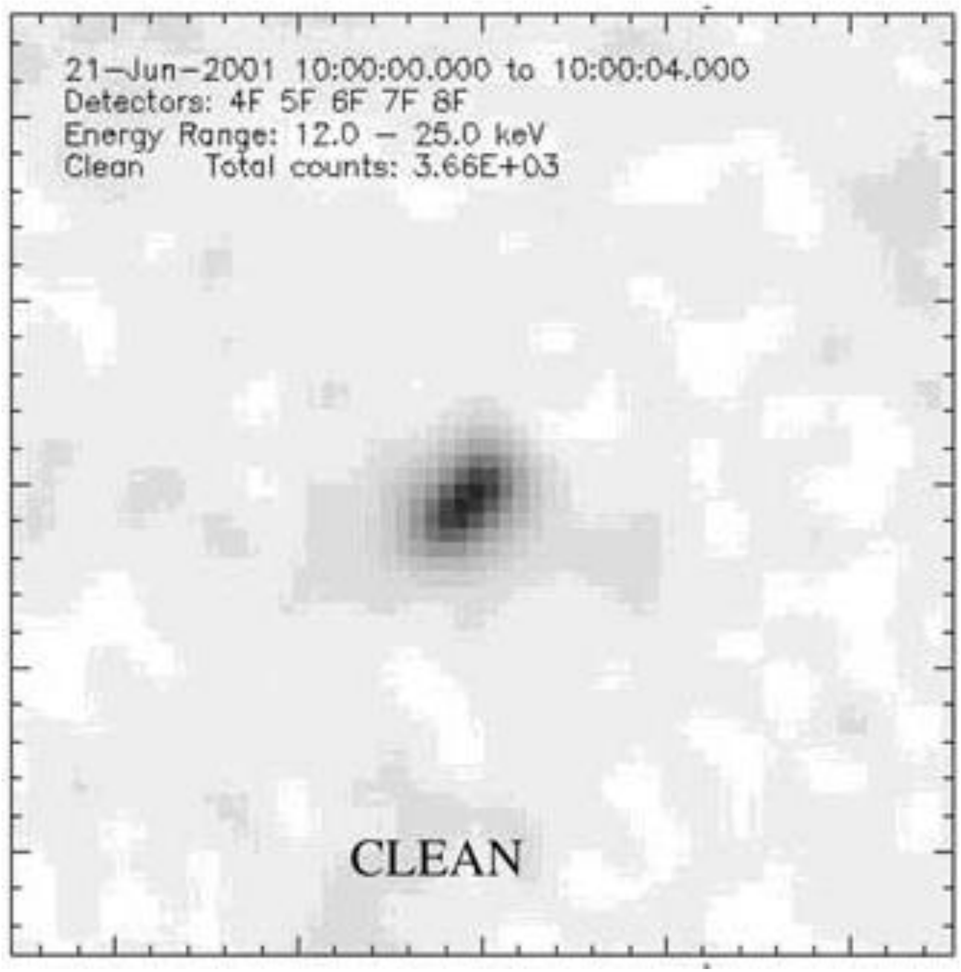
Clean assumes that the image is a superposition of point sources convolved with **Point Spread Function** (**PSF**) and iteratively cleans the initial back-projected image until either negative residuals or max number of iterations Högbom (1974).

Advantages:

- Relatively fast
- Positive fluxes

Disadvantages:

- Nonlinear
- Overestimates source sizes
- Likely to misinterpret extended sources



Hurford et al, 2002

RHESSI imaging: MEM

Maximum Entropy Methods (MEM) algorithm looks for the map that is both consistent with the data and contains the least information about the source (i.e. maximum entropy). (Hurford et al, 2002)

$$\chi^2 = \sum_i \frac{(\mathcal{C}_i - \mathcal{E}_i)^2}{\sigma_i^2}, \qquad \mathcal{H} = -\sum_m F_m \log F_m,$$

consistency with data

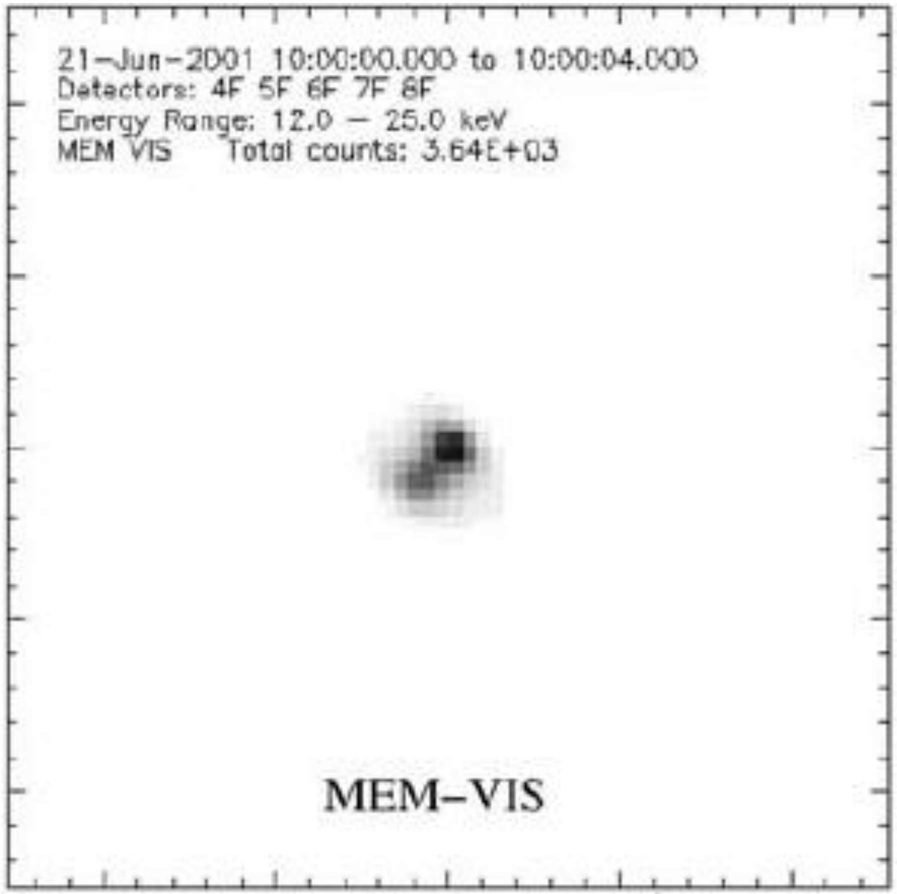
Maximising entropy

Advantages:

- Relatively fast
- Positive fluxes

Disadvantages:

- Nonlinear
- Underestimates source sizes



Hurford et al, 2002

RHESSI imaging: Forward Fitting

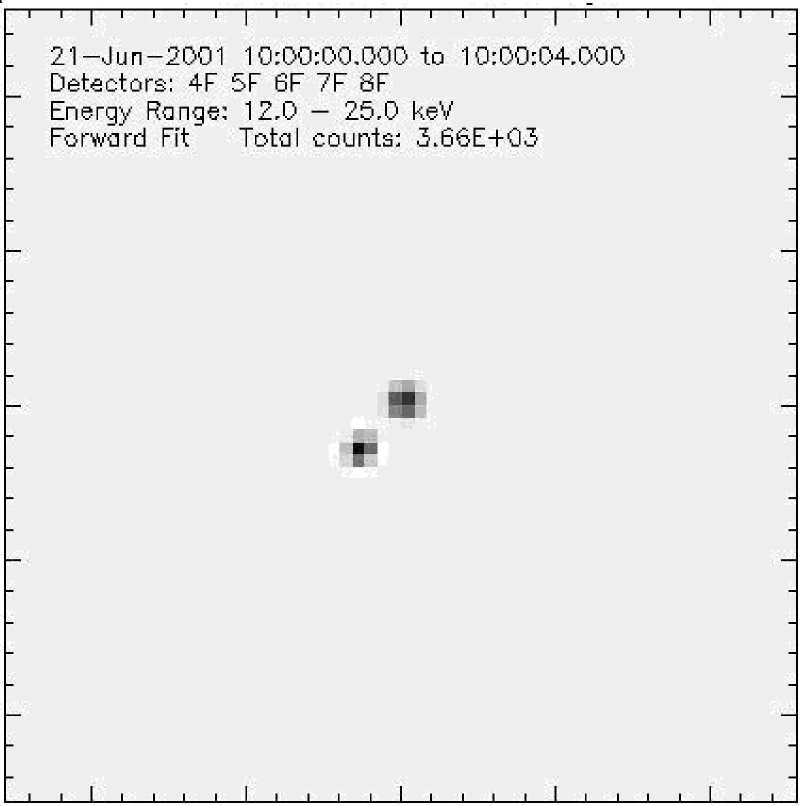
Forward Fit assumes that the source can be presented as a prescribed functional form with a number of free parameters (for example, two circular Gaussian sources) and looks for the parameters which produce a map that is consistent with the data. (Aschwanden et al, 2002; Hurford et al, 2002)

Advantages:

- Fast
- Positive fluxes
- Source sizes and errors on parameters

Disadvantages:

- Does not work well for complex sources



Hurford et al, 2002

RHESSI imaging: PIXON

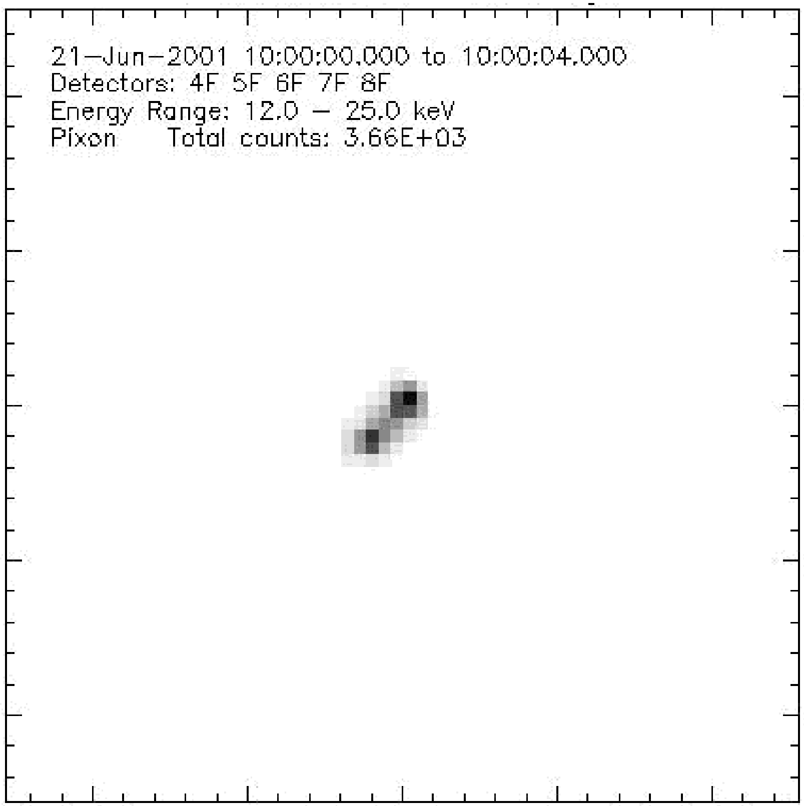
Pixon looks for **the simplest model** for the image that is consistent with the data (suitable CHI2) (Puetter, 1995; Metcalf *et al.*, 1996). Pixon simultaneously minimises smoothes the image *locally (minimisation of independent patches)* and simultaneously consistent with the data.

Advantages:

- Photometric accuracy
- no spurious features

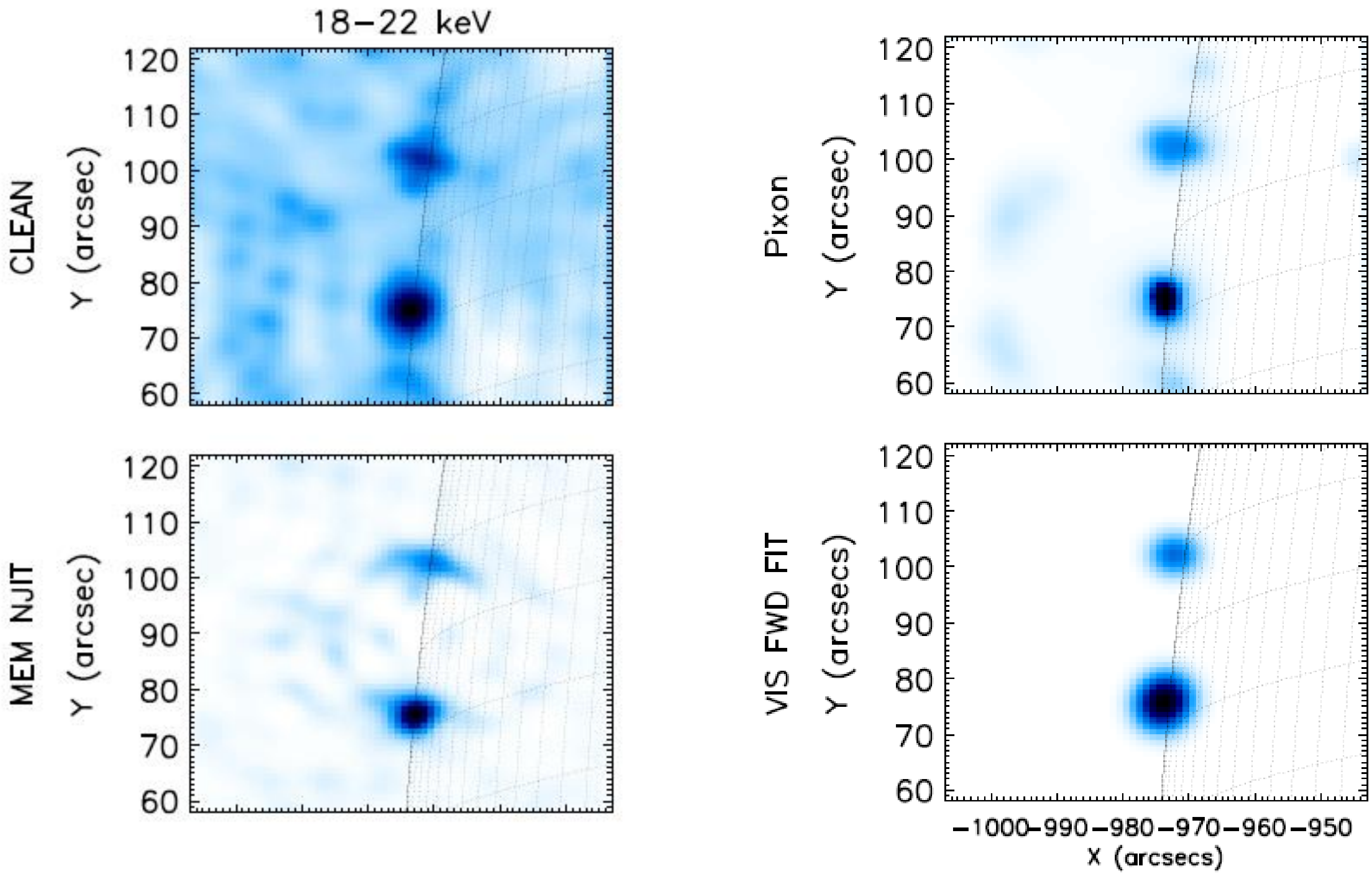
Disadvantages:

- Very slow (too orders of magnitude)



Hurford et al, 2002

Comparing imaging algorithms



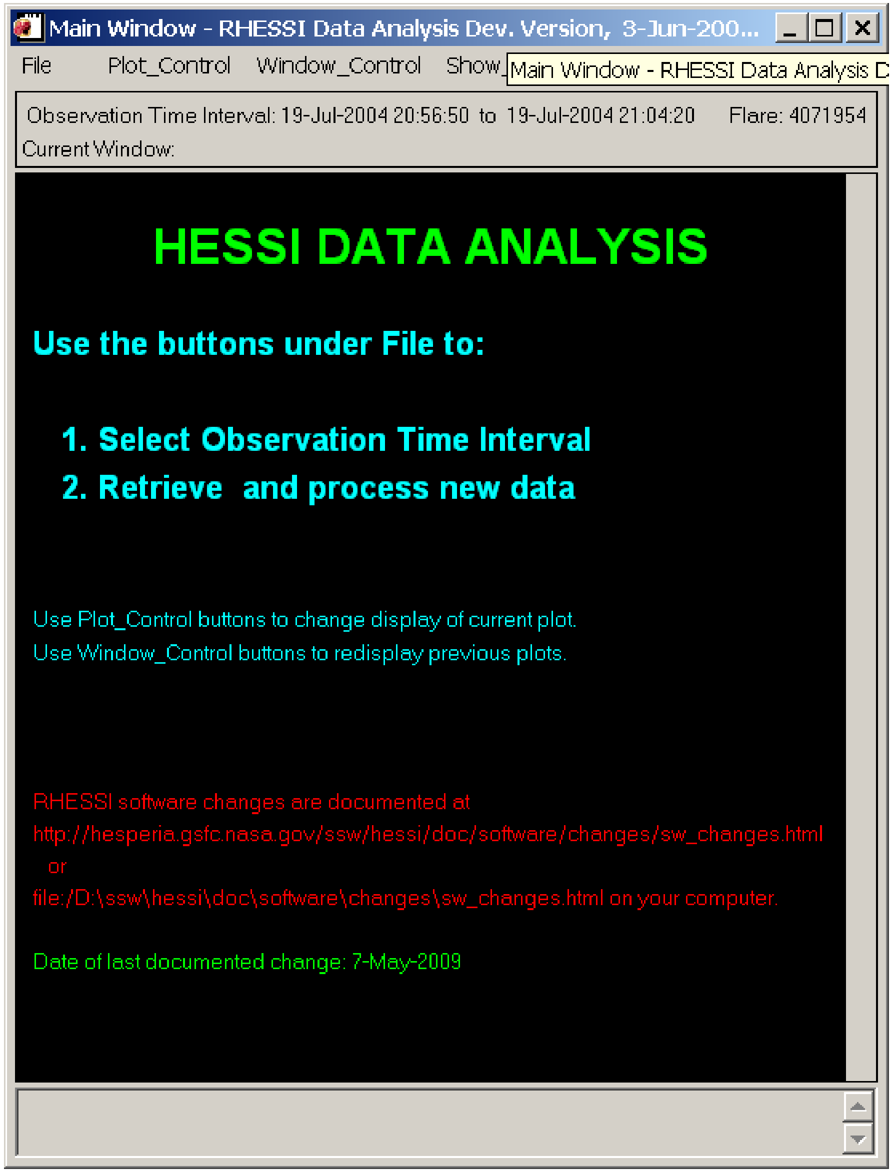
06 - January, 2004 flare

RHESSI imaging

1) Start GUI interface

IDL>hessi

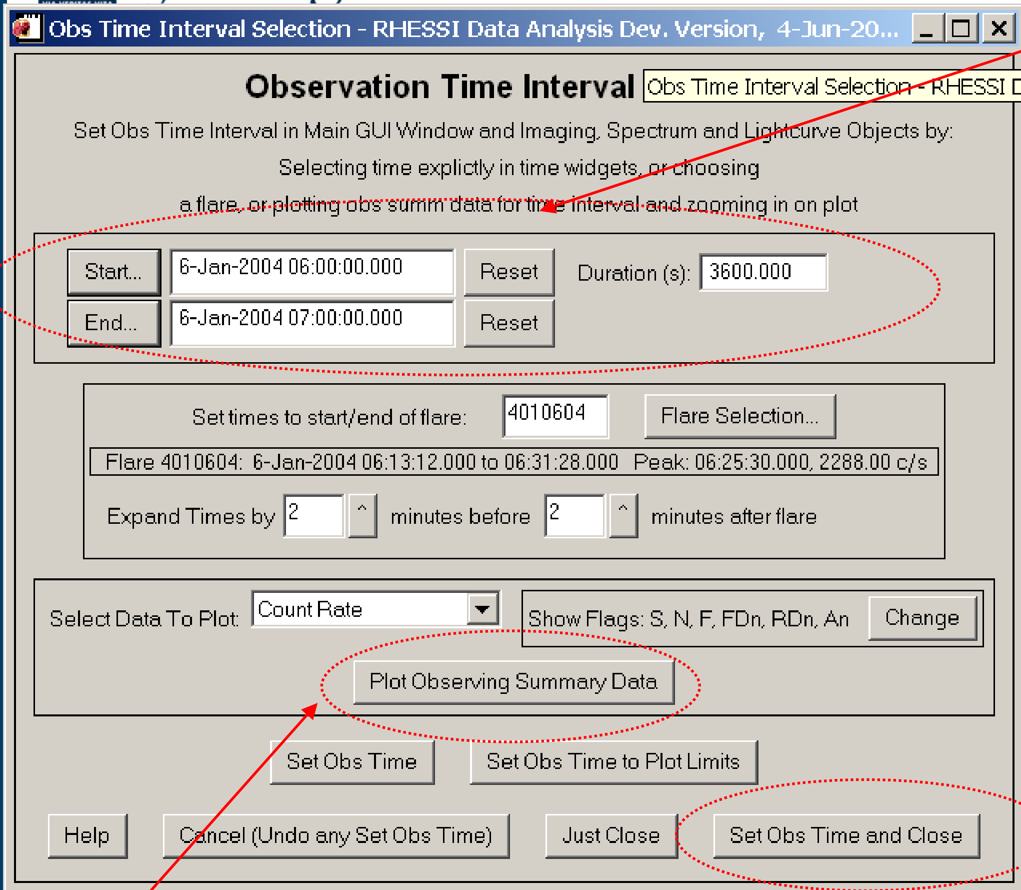
2) You will see this



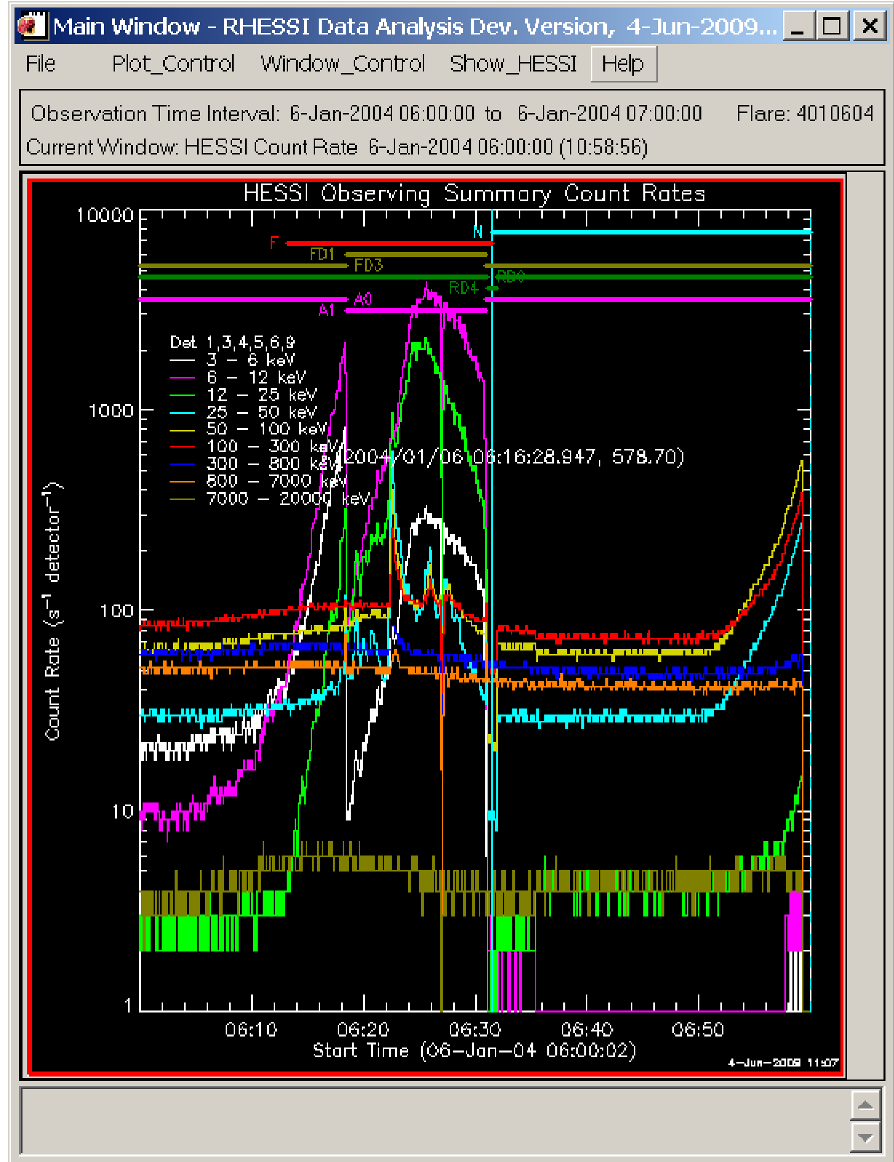
3) Let us select a flare

January 06, 2004 ~06:20 UT

Observing summary data



1) Select time interval



2) Plot observing summary data

3) Set obs time & close

RHESSI imaging

In the mane window select
File ->Retrieve/Process data-> Imaging

1) Select time interval

2) Select energy range

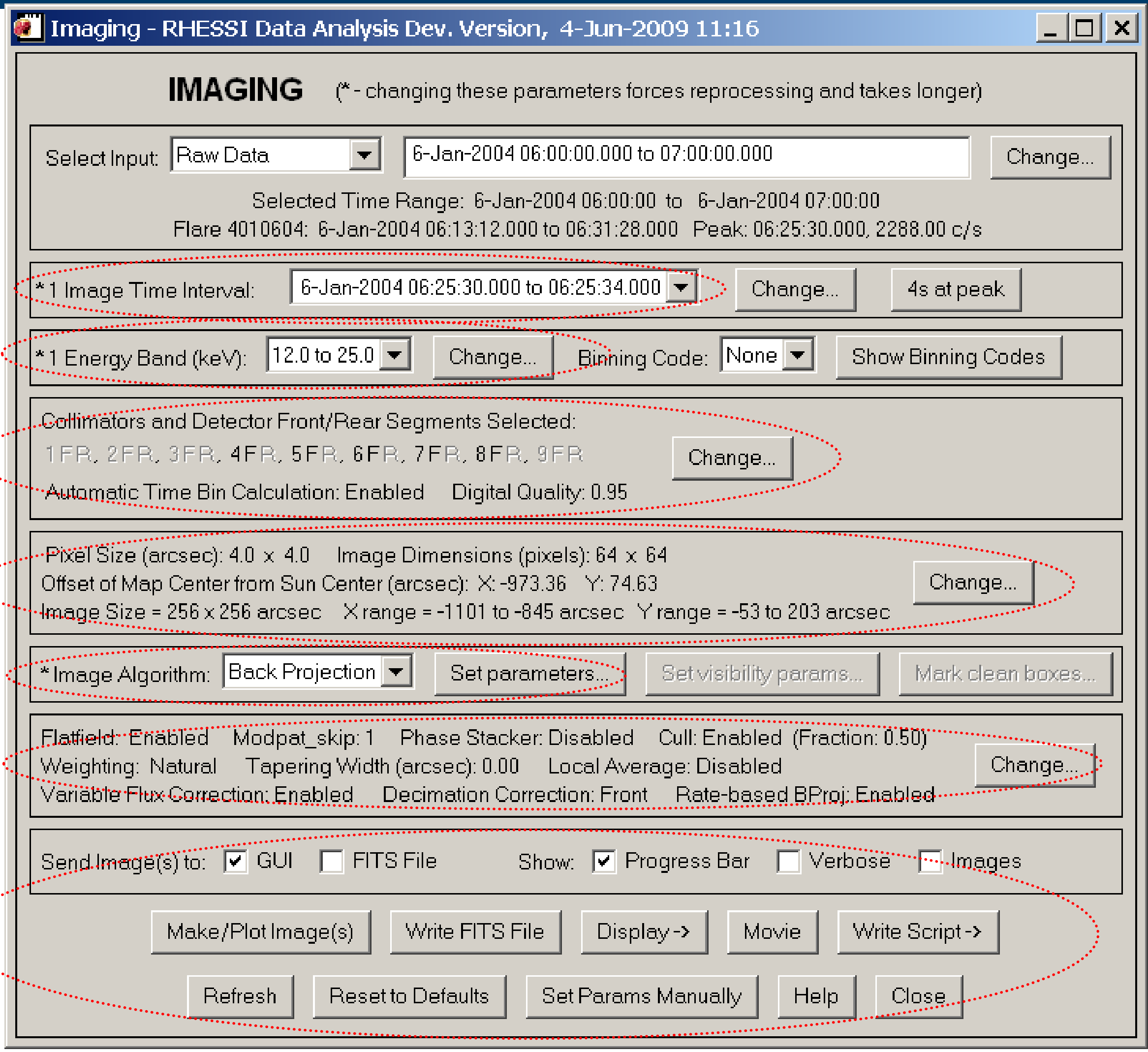
3) Select detectors

4) Select image details:
size, pixel size, xy offset

5) Select image algorithm

6) Do not mess with this
(unless you know what you are doing)

7) Choose output



Imaging time interval

Selecting time interval….

1) Define time interval here

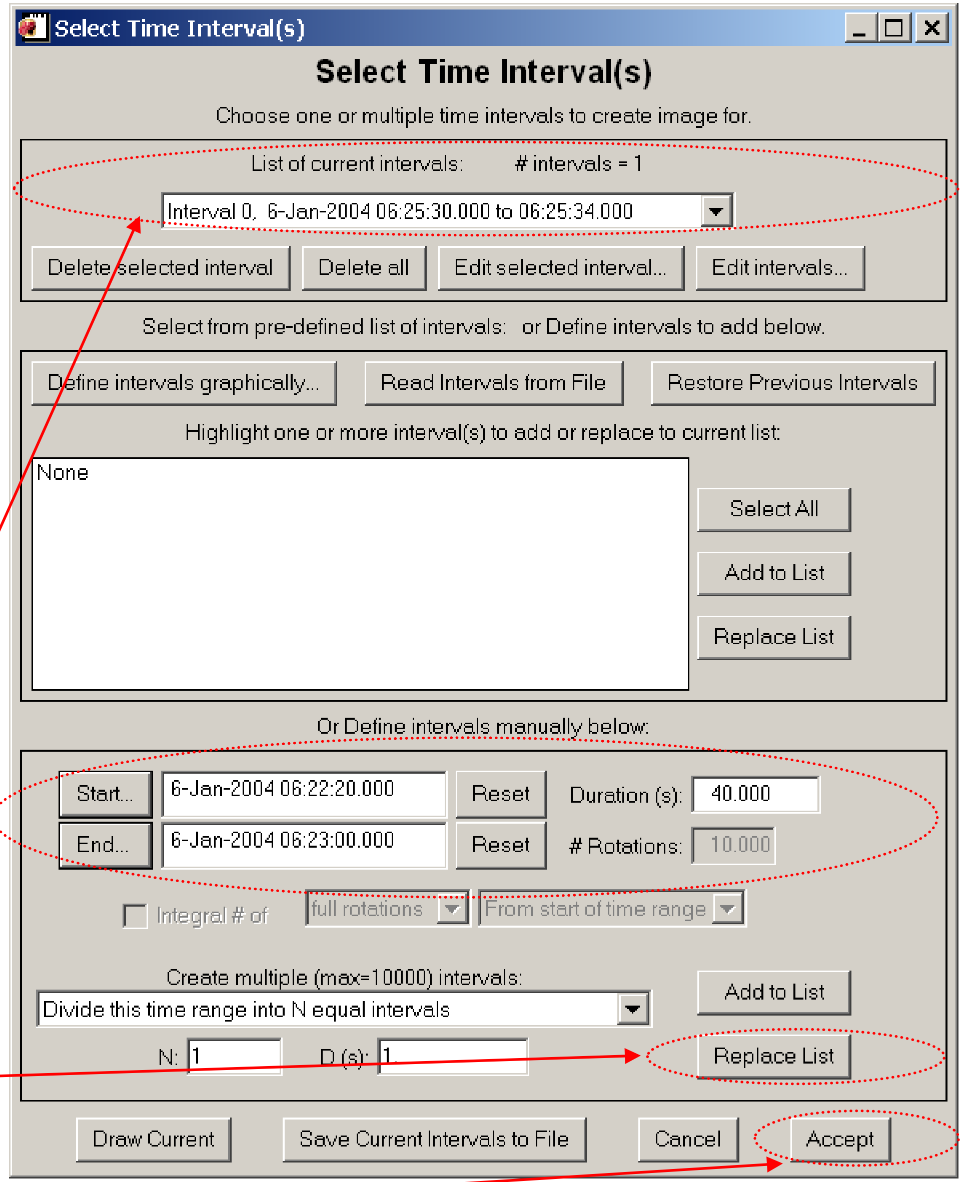
Good practice:

a) Time interval is a multiple of spin period (or a half)

b) The weaker the flare the longer the time (20-40 seconds => good images for a medium class flare)

2) Always press replace list
(*note the change at the top*)

3) Press accept



Imaging energy range

Selecting energy interval….

1) Define energy range here

Good practice:

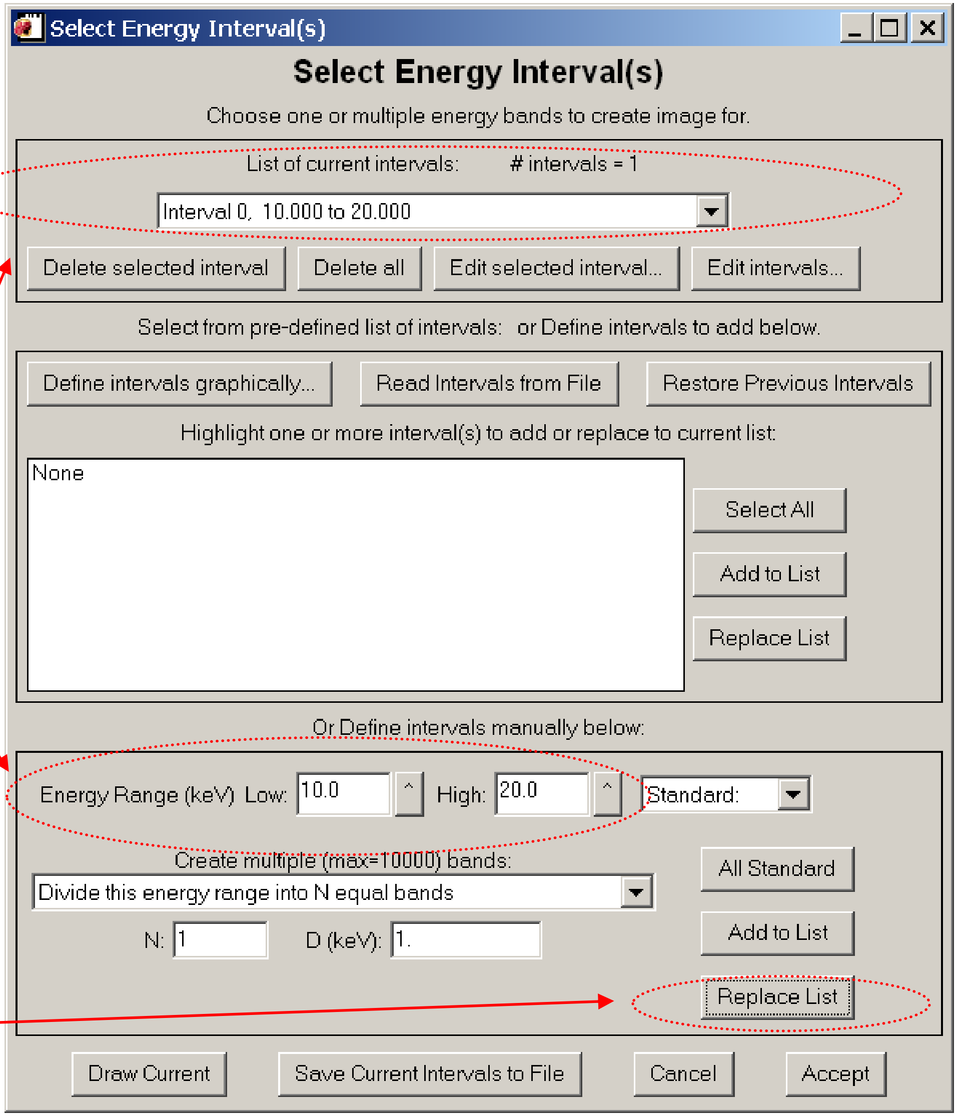
a) Time interval is more than a few keV

b) The weaker the flare the longer the energy range (5-20 of keV is OK)

c) The higher the energy the longer the energy range (recall X-ray spectrum) (~100 of keV at 200 keV)

2) Always press replace list (*note the change at the top*)

3) Press accept



Selecting Collimators/Detectors …

1) Define RMC here

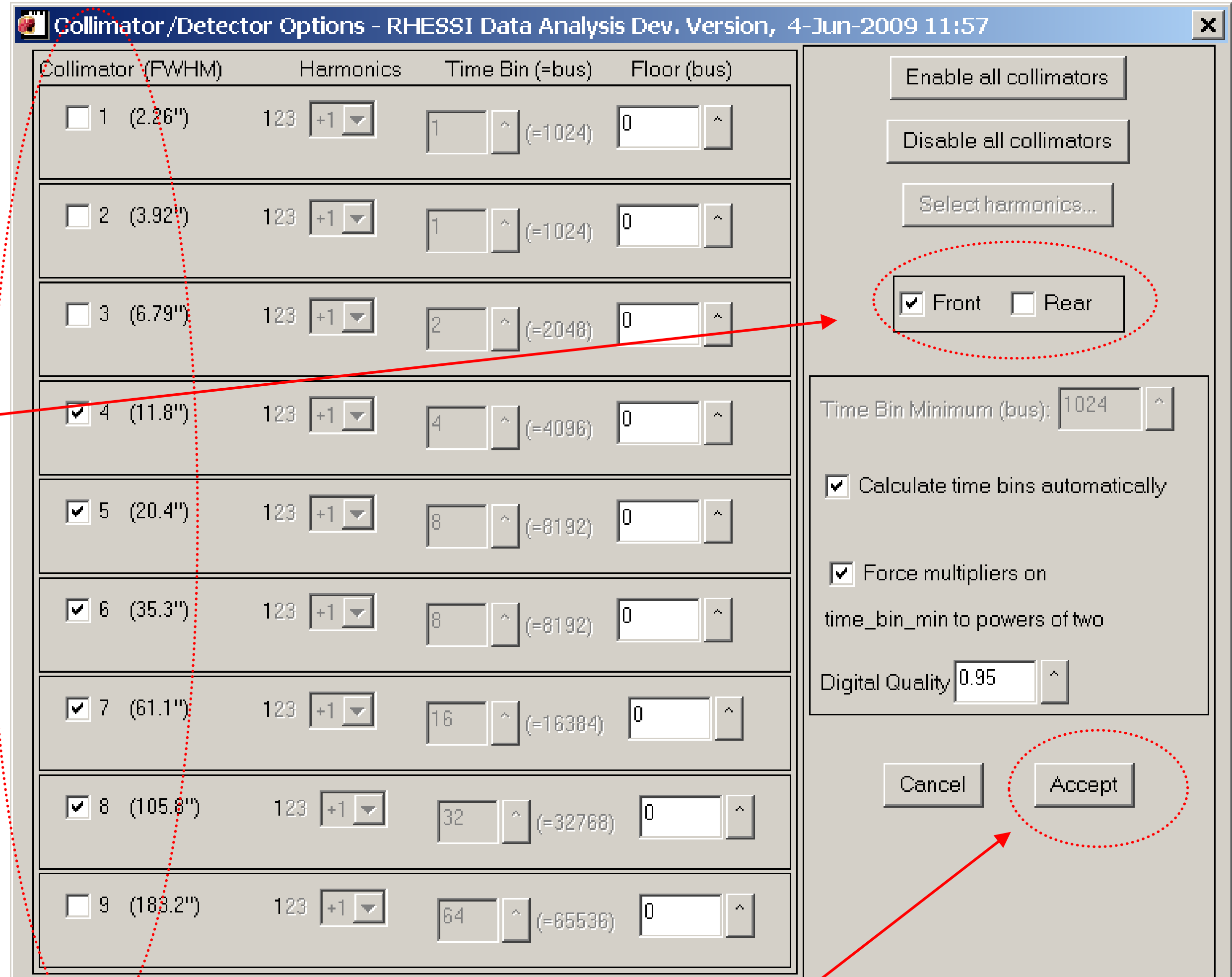
Good practice:

a) **Front** for energies <~300keV, **Rear** for higher energies (Det#1 up to 100 keV)

b) Det #3-8 is a common choice

c) Det#2 above 20-25 keV only

d) Det#9 for scales >~180"



2) Press accept

Image/pixel sizes

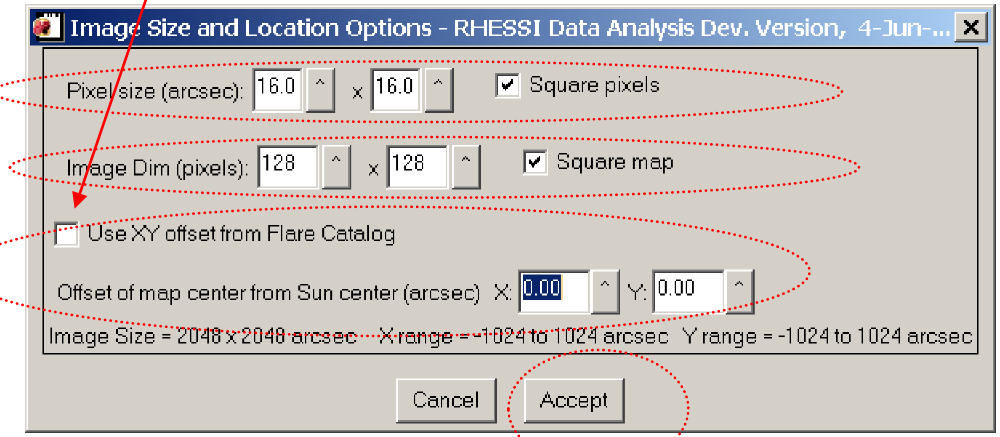
Selecting Collimators/Detectors …

It is not bad idea to use XYoffset from flare catalog (put tick if you want it)

1) Define pixel size

2) Define image size

3) Set the centre of the image as offset from the Sun centre



4) As usual press ‘accept’ when finished

Good practice:

a) Pixel size less than RMC FWHM

b) Use small image size ~64x64 or 32x32 (especially with Pixon)

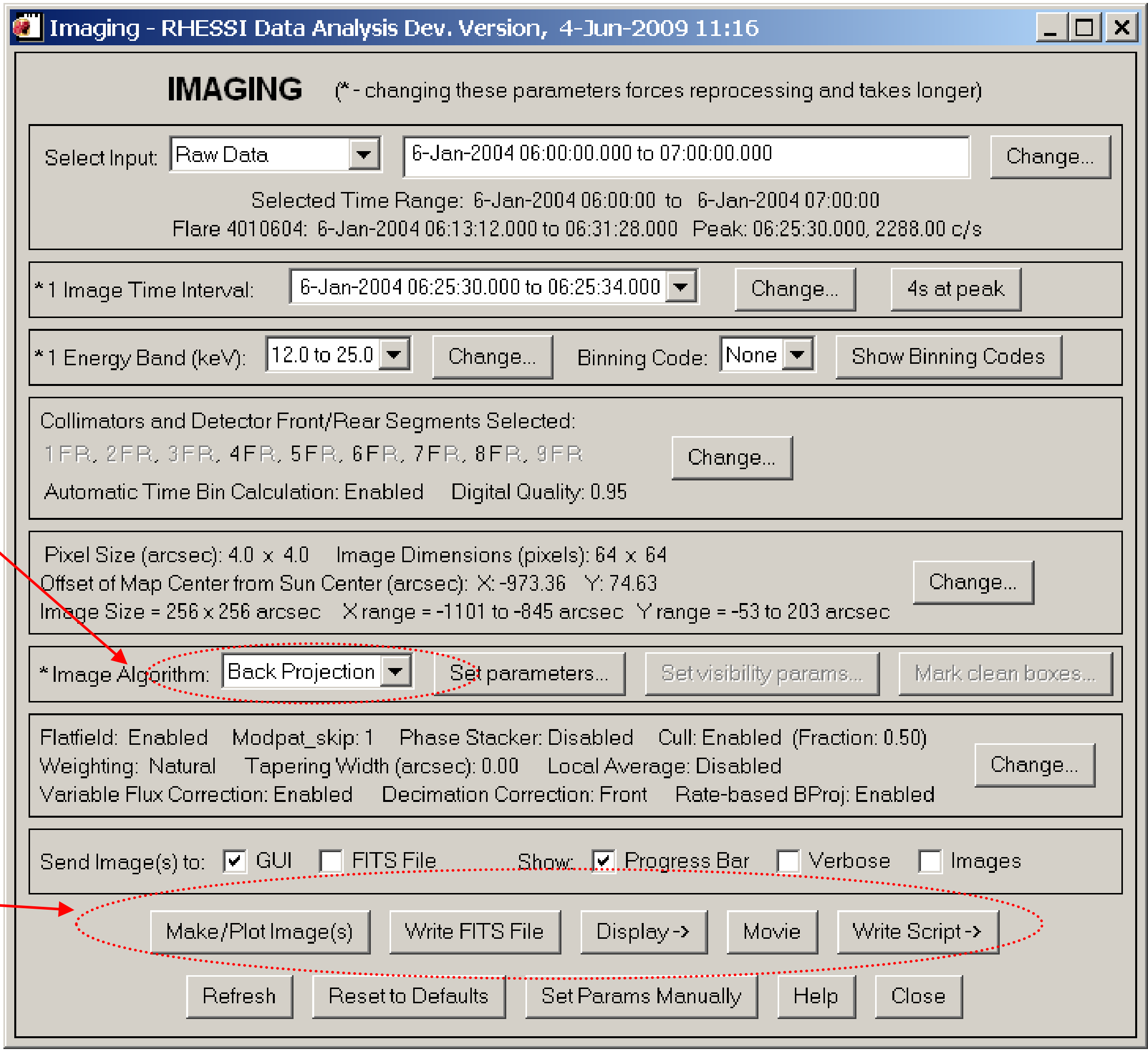
c) Keep in mind where the spin axis is

RHESSI imaging

Finally …

1) Choose algorithm

2) Output type (save as image as a file or display in the main window)



Synoptic data

Use synoptic data … (almost all main solar data are accessible !!!)

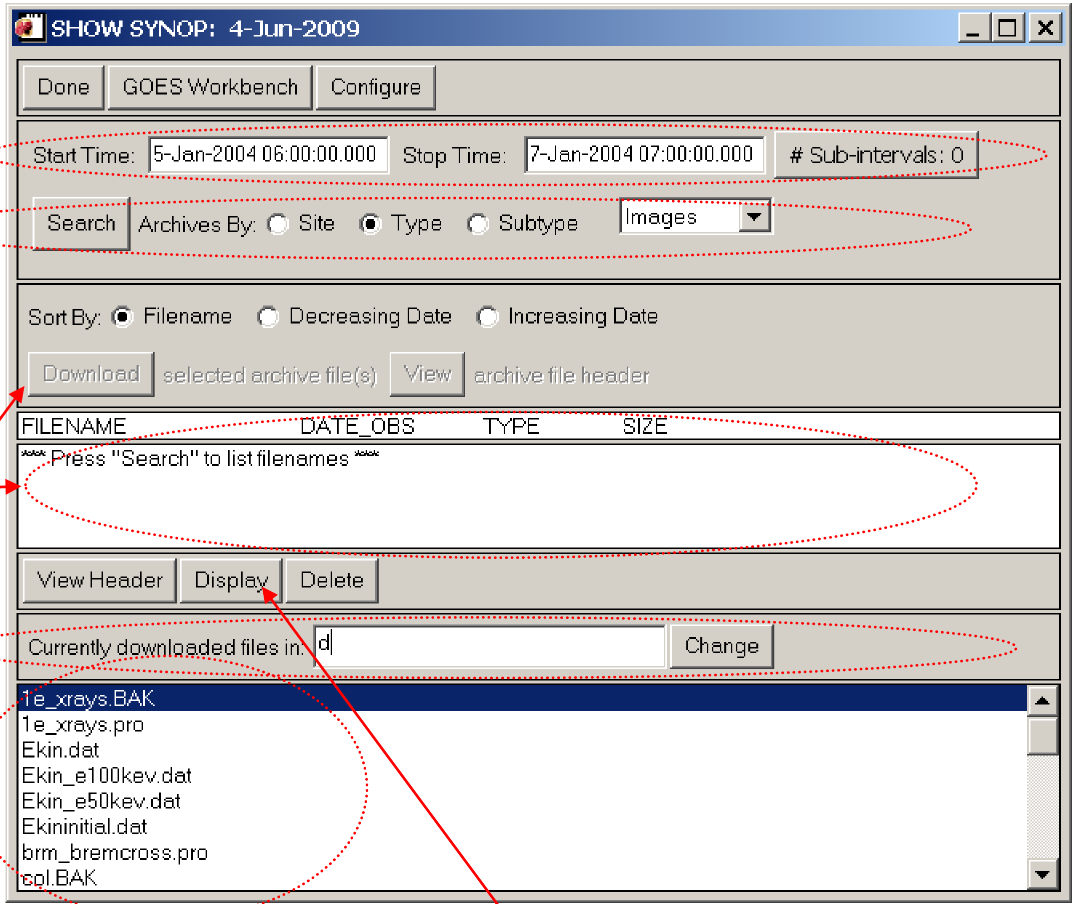
1) Choose the time range

2) Choose type of data
and press ‘**search**’

Search results will appear here

3) Set your local folder

4) Download the data



Download results will appear here

4) Plot the data

Making an image from command line

```
obj = hsi_image()                                                         <= Defining imaging object anything instead of 'obj'
obj-> set, det_index_mask= [0, 0, 1, 1, 1, 1, 1, 1, 0]                    <= Detectors used #3-8
obj-> set, im_energy_binning= [10.0, 20.0]                                <= Energy range used from 10 to 20 keV
obj-> set, im_time_interval= ' 6-Jan-2004 '+['06:22:20', '06:23:00']      <= time interval
obj-> set, image_algorithm= 'Back Projection'                             <= image algoritm; could be Clean, Pixon, etc

obj-> set, image_dim= [128, 128]                                          <= image size in pixels
obj-> set, pixel_size= [32., 32.]                                         <= pixel size in arcseconds
obj-> set, use_flare_xyoffset= 0                                          <= if set to 1 uses catalog data, if set to 0 not
obj-> set, xyoffset= [0.0, 0.0]                                           <= sets image centre coordinates

data = obj-> getdata()     ; retrieve the last image made



obj-> plot                 ; plot the last image
obj-> plotman              ; plot the last image in plotman
```

Visibilities forward fit 1

```
obj = hsi_image()
obj-> set, det_index_mask= [0, 1, 1, 1, 1, 1, 1, 0, 0]
obj-> set, im_time_interval= '6-Jan-2004 '+['06:22:20', '06:23:00']
obj-> set, im_energy_binning=[25.,35.]

obj-> set, image_algorithm= 'VIS FWDFIT'        <=visibilities forward fit to be used
obj-> set, xyoffset=[-970,73]                    <= centre of the map
obj-> set, use_phz_stacker= 1                    <= stacking into roll bins
;obj-> set,PHZ_radius=10.                        <= automatic choice of roll bins based on the characteristic scale 10”
obj-> set,image_dim=[64,64]
obj-> set,pixel_size=[1.0,1.0]
obj-> set, vf_multi=1                            <= two sources

obj-> set,phz_n_roll_bins_control=[20,38,20,20,12,12,12,12,20]    <= manual choice of roll bins
;sets the number of roll bins per detector

; useful controls when calculating the visibilities
obj-> set, vis_edit=1, vis_combine=1
; remove outliers and combine conjugates
```

Visibilities Forward Fit 2

our_fit=obj ->get(/vf_srcout) <= gets parameters out of object
;gets visibilities parameters

fit_err=obj ->get(/vf_sigma) <= errors on the fit parameters
;gets visibilities parameter errors

;you can set some initial parameters for you visibility structure
our_fit.srctype=['ellipse', 'ellipse'] <= two ellipse fit
our_fit.srcflux=[11,9] <= fluxes per ellipse
our_fit.srcfwhm=[7,6] <=FWHMs of the sources
our_fit.eccen=[0.7,0.7] <= ellipse eccentricities
our_fit.srcpa=[175,-178] <= position anlgles
our_fit.srcx=[907,904] <= x coordinates of the ellipse/circle centre
our_fit.srcy=[248,272] <= y coordinates of the ellipse/circle centre

<= two source sizes per

obj -> set, vf_srcin=our_fit <= setting parameters as initial guess for forward fit
; sets out fit params

obj-> plot
obj->plotman <= plotting the output map (image)

Instead of conclusions: useful URLs

These lecture notes and example IDL scripts:
http://www.astro.gla.ac.uk/users/eduard/sodas

RHESSI imaging overview (good collection):
http://hesperia.gsfc.nasa.gov/hessi/instrumentation.htm

RHESSI imaging tutorials (from first steps to advanced level):
http://hesperia.gsfc.nasa.gov/rhessidatacenter/imaging/overview.html

Description of all RHESSI imaging software parameters:
http://hesperia.gsfc.nasa.gov/ssw/hessi/doc/hsi_params_all.htm