Where Do AFNI Datasets Come From?

- Method 1: Create datasets with program to3d [principal subject of this talk]
  - ⋆ Inputs are arrays of numbers — image files
- Method 2: Realtime input from an external image source program (e.g., directly from the scanner's reconstructed images)
  - ⋆ AFNI program Imon reads image files from GE realtime EPI output, checks them for various errors, sends them into AFNI for display and formatting — while acquisition continues
  - ⋆ Sample program rtfeedme can be used to write your own image source program
- Method 3: AFNI programs can read other formats for display and analysis
  - ⋆ ANALYZE™ 7.5 format — `.hdr/.img` file pairs
    - ↪ Used by SPM and many other programs
    - ↪ Major drawback: lack of spatial orientation and position information in header $\Longrightarrow$ can be difficult to overlay ANALYZE datasets with other datasets
  - ⋆ MINC format — `.mnc` files
    - ↪ Used by software from Montreal Neurological Institute — mnitools
  - ⋆ CTF format – `.svl` files
    - ↪ Generated from CTF MEG data analysis software package

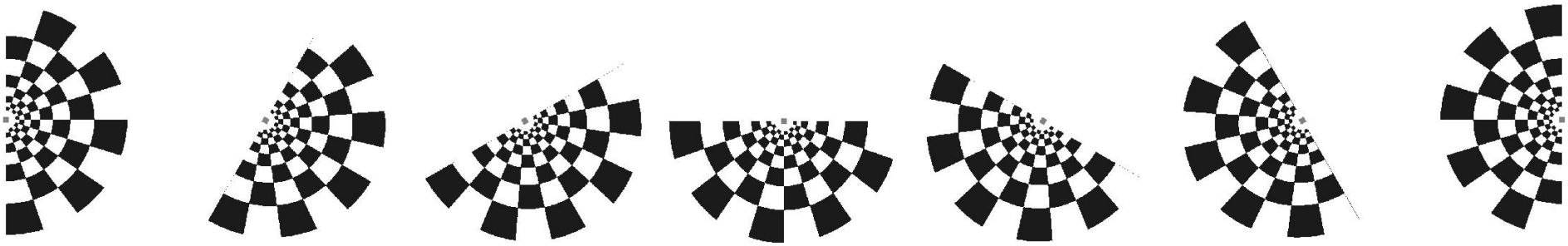
- ⋆ Dataset stored as columns of ASCII-formatted numbers — `.1D` and `.3D` files
  - ↪ Used to store datasets when knowing where the data points are in space isn't important for the analysis
  - ↪ Example: node-wise analysis of group data on surfaces
  - ↪ Each column corresponds to one sub-brick
  - ↪ Each row corresponds to one voxel or node
  - ↪ .1D files: just columns of numbers
  - ↪ .3D files: contain an XML header with geometrical information
- ⋆ NIFTI-1 format — `.hdr/.img` file pairs or `.nii` files
  - ↪ New format, modified from ANALYZE 7.5 to include more information
  - ↪ Supposed to be mostly compatible with ANALYZE 7.5 compatible programs
  - ↪ Format just being finalized now [late 2003]; will be supported by SPM, AFNI, FSL, Brain Voyager

- Method 4: Output of most AFNI programs is AFNI-formatted datasets — `.HEAD/.BRIK` file pairs
  - ⋆ AFNI utility programs exist to re-write AFNI-formatted datasets into ANALYZE, MINC, and .3D formats
  - ⋆ In the future, AFNI programs will be able to write out NIFTI-1 .nii formatted datasets directly

Creating AFNI Datasets with Program to3d

- to3d reads image files — each containing 1 or more 2D slices — and assembles them into AFNI datasets
- The collection of all the 2D slice data forms the `.BRIK` file
  - ⋆ An AFNI dataset can contain a single slice
- You must also provide to3d with some auxiliary data (for the `.HEAD` file):
  - ⋆ Orientation of the slices in space
  - ⋆ Size of the slices or of the voxels
  - ⋆ Slice offset — where is the dataset volume located in space?
  - ⋆ For 3D+time datasets, also need slice timing information
  - ⋆ to3d 'knows' how to get some of this auxiliary information from image file headers for some image file formats:
    - ↪ ANALYZE 7.5 `.hdr/img` pairs contain voxel size information
    - ↪ Siemens `.ima` image arrays contain voxel size and orientation information
    - ↪ GE `I.` files contain voxel size and orientation information
    - ↪ DICOM files contain lots of relevant information
      - ▷ Manufacturers' variations on DICOM are frustrating

- to3d runs in two modes:
  - ⋆ Command line mode: you provide all auxiliary information on command line
    — Useful to get things done fast, and for putting into scripts
  - ⋆ Graphical user interface (GUI) mode: you provide auxiliary information by filling out an on-screen form
    — Useful for explaining concepts, and for AFNI neophytes
- Sample #1: data from NIH GE 3 Tesla scanner
  - ⋆ Files stored in archive `AFNI_sample_01.tgz`
    — unpack with command `gzip -dc AFNI_sample_01.tgz | tar xf -`
    — unpacks into directory `AFNI_sample_01/`
  - ⋆ Anatomical (SPGR) data ⇒ 3D dataset (no time; 1 sub-brick)
    — 124 axial slices in subdirectory `SPGR_anat`
  - ⋆ Functional (EPI) time series data ⇒ 3D+time dataset (160 sub-bricks)
    — 2880 images (18 coronal slices, 160 reps) in subdirectory `EPI_run1`
    — Visual stimulation task: rotating hemifield flashing checkerboard



- Experiment log, taken at scanner:

NIMH-LBC-METHODS FMRI RESEARCH — **RUN DATA**

ExpCode: AR25 Subject: [redacted] 30-18-08-8 Date: April 20 01 [redacted] 8AM.

Protocol#: P[redacted] Start Time (24-hr): 9AM - _ Investigator: Shruti Tech: ZSS

Scanner: 3T Coil: Wong / MAI / GE

BW 15.63

Anat Scan 1 Type: SPGR / FSE / MPIR / IR FSPGR TE(ms): M=Full TR(ms): _____ Flip: 17° NEX: _____ FOV(mm): 240

Matrix: 256 X 256 #Slices: 124 Plane: Ax / Cor / Sag Thickness(mm): 1.1 First: 150.6 Last: 84.7 Taken last

Anat Scan 2 Type: SPGR / FSE / MPIR / _____ TE(ms): _____ TR(ms): _____ Flip: _____ NEX: _____ FOV(mm): _____

Matrix: _____ X _____ #Slices: _____ Plane: Ax / Cor / Sag Thickness(mm): _____ First: _____ Last: _____

EPI-Scan: GE-EPI / SE-EPI / GE RT EPI TE(ms): 25 TR(ms): 2 Flip: 90 NEX: _____ VTE: _____

Plane: Ax / Cor / Sag FOV(mm): 240 Matrix: 64 Thickness(mm) 4 #Slices: 18 First: P96.8 Last: P28.8

Response Data: None / Accuracy / RT **Other Data:** EKG / HR / BP / RR / CO2 / GSR / EEG / EP / EMG / NONE _____

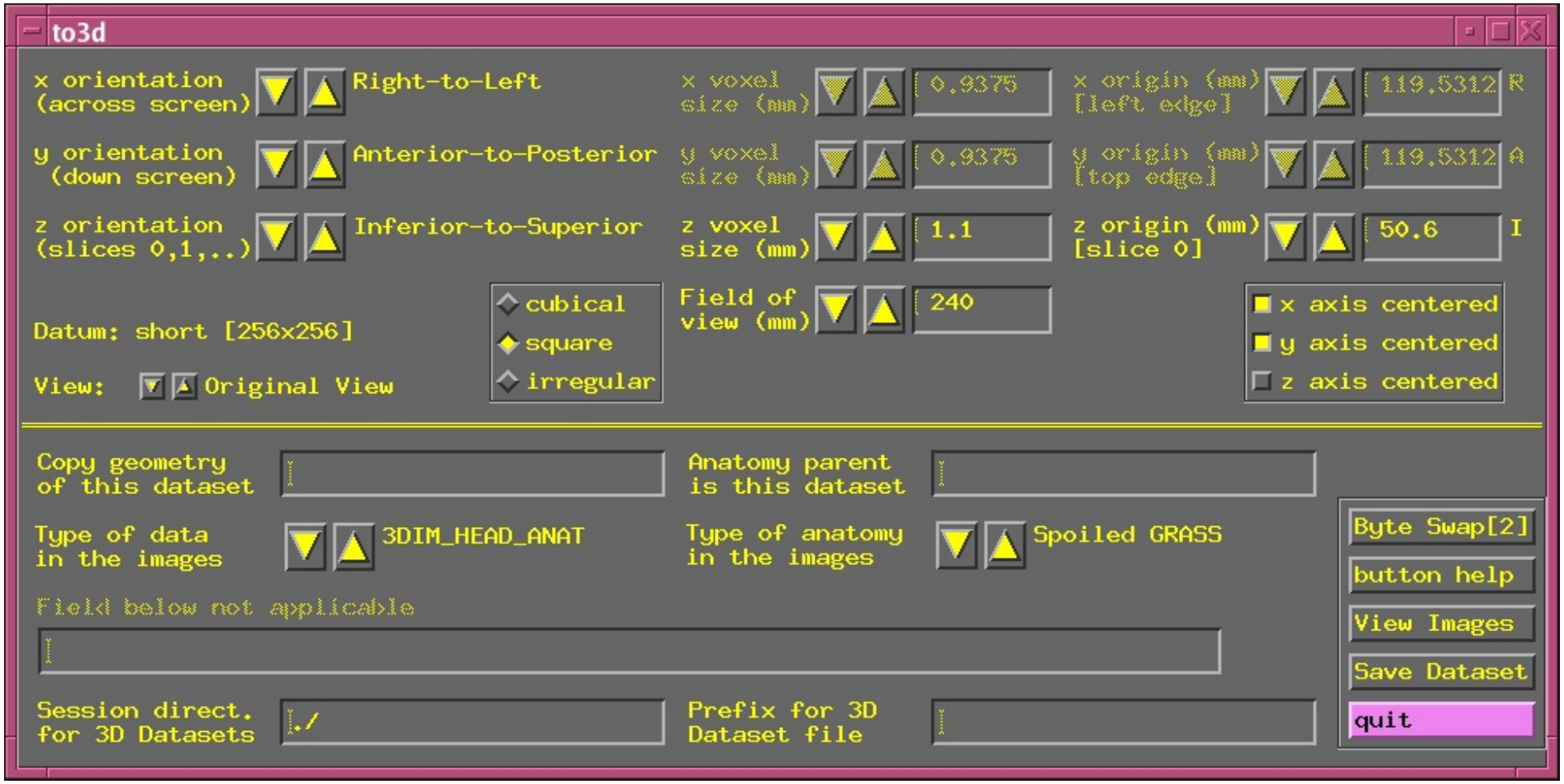
Timing: #Reps: _____ Skip#: _____ #Cycles: _____ #Conditions/Cycle: _____ #Reps/Condition/Cycle: _____

Other Timing: 180° wedge 30 ON 15 OFF 270° total 30s pre 20s post ⇒ T.O.P

(38s on command line, 8sec dropped)

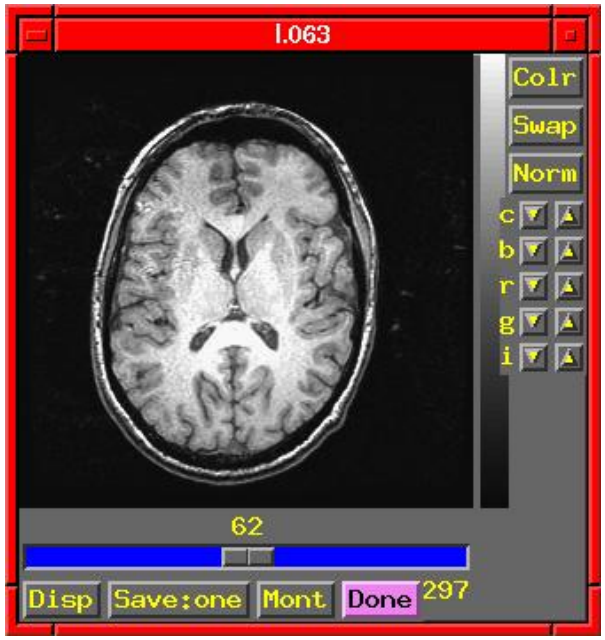
Run#	Time	Conditions	Stimulus File	Data File	Response File	
___	___	180° wedge, CCW	003/I.019 → 043/1900	___	___	___
___	___	CCW	43/901 → 103/783	___	___	___
___	___	CW	103/784 → 163/666	___	___	___
___	___	CW	163/667 → 223/549	+ may have been ~ 1sec starting scan ...	___	___
___	___	CCW	223/550 → 283/432	___	___	___
___	___	CCW	283/433 → 343/315	___	___	___
___	___	CW	343/316 → 403/198	___	___	___
___	___	CW	403/199 → 463/081	___	___	___

- Using to3d to assemble the SPGR dataset:
  - ⋆ `cd AFNI_sample_01/SGPR_anat` — change directory, to get at images
  - ⋆ `ls` — to see what files are there (should see files `I.001` ... `I.124`)
  - ⋆ `to3d I.*` — run to3d, reading in all the image files — GUI pops up:



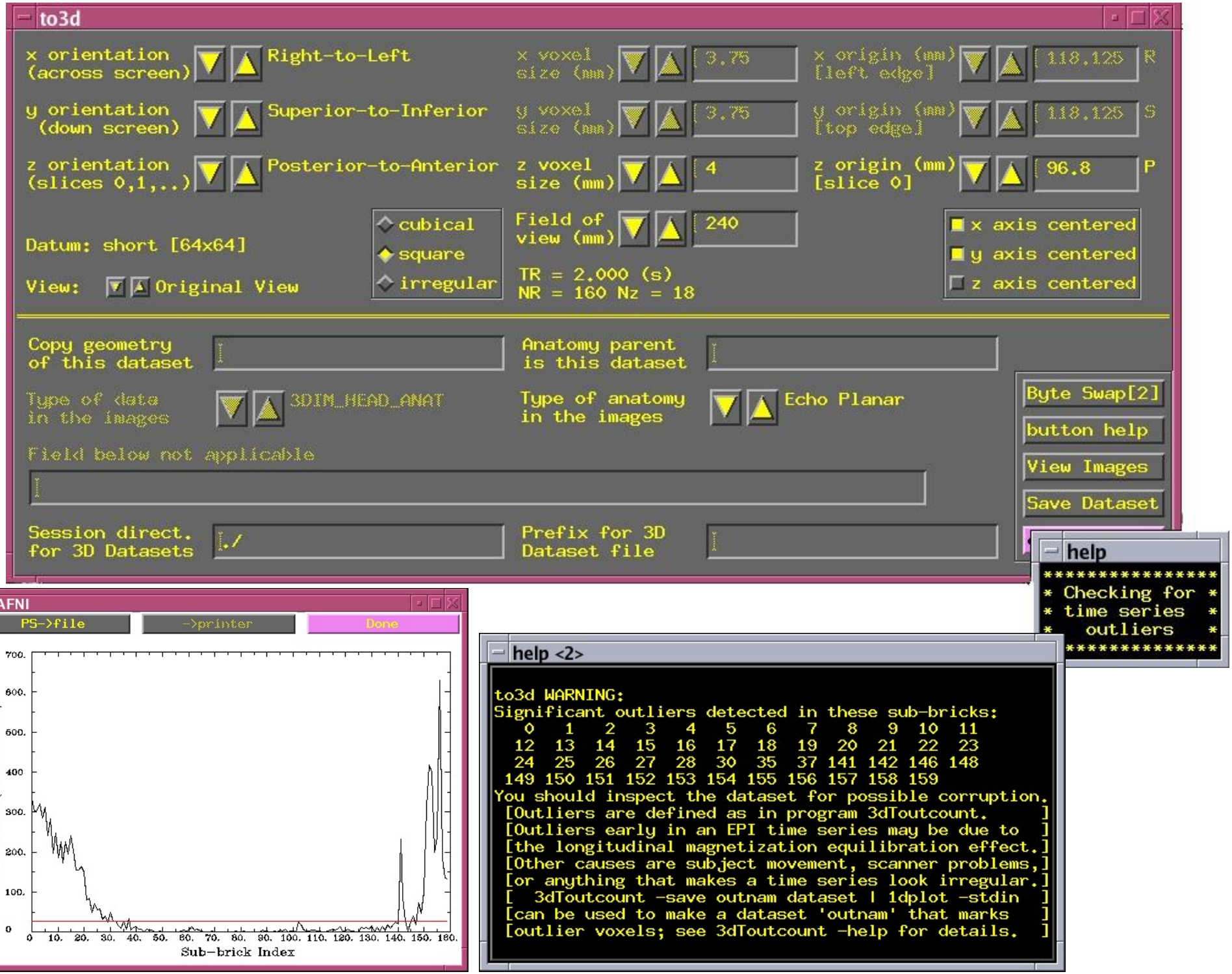
  - ↪ to3d understands GE I.* files, and so has filled in some of the GUI
  - ↪ Note: z origin field 50.6 I corresponds to experiment log

- To check images that were just input, click the `View Images` button in the to3d form
  - ↪ Window is the same as the AFNI image viewer
  - ↪ Slider below image lets you move between slices

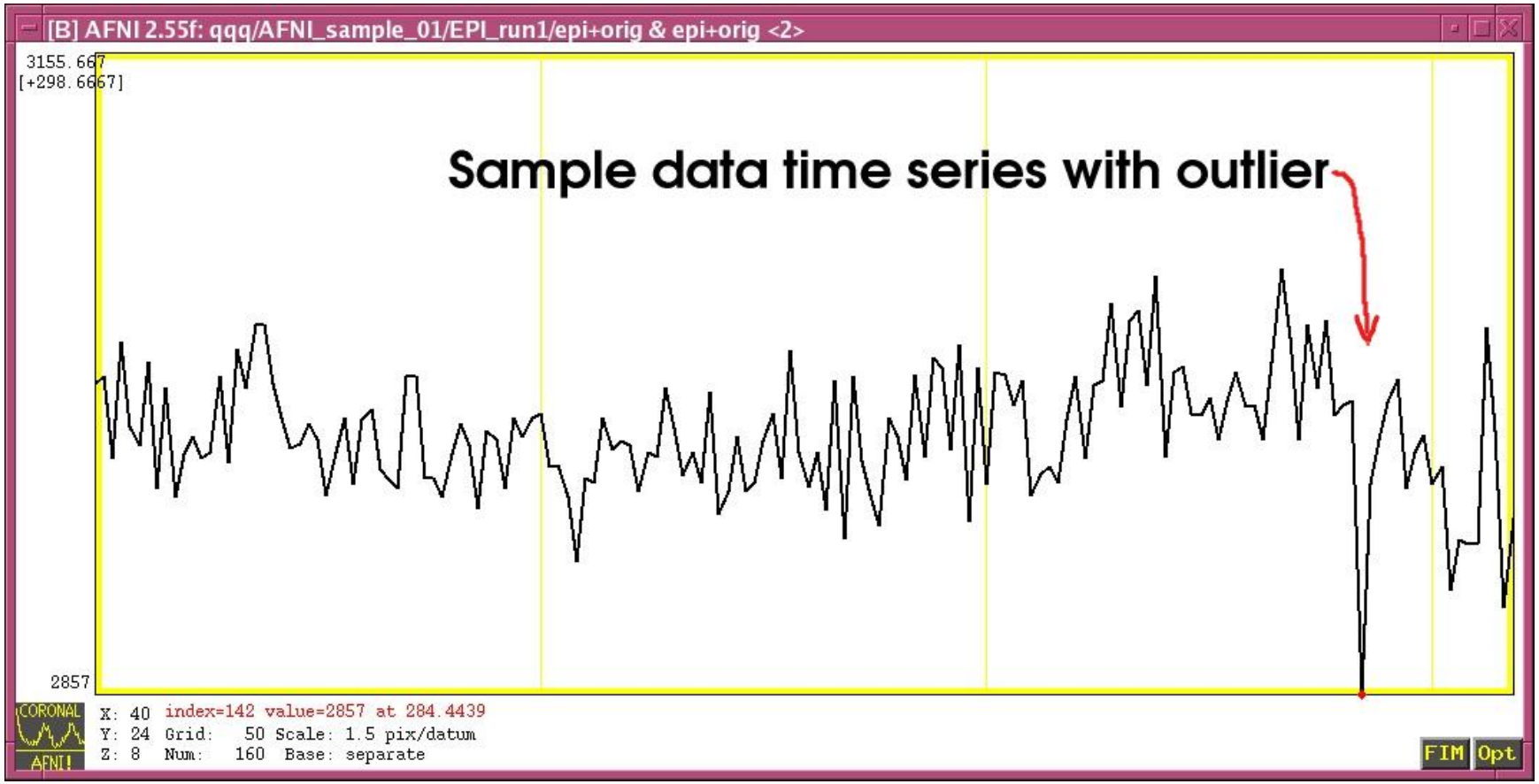


- In this example, to3d has all the information needed from the I.* headers
  - ↪ All you need to do is supply the dataset `Prefix`, then press `Save Dataset`
    - ▷ look at the bottom right of the to3d GUI for these controls
    - ▷ I suggest the prefix `anat`
  - ↪ Dataset files `anat+orig.HEAD` and `anat+orig.BRIK` will be created
  - ↪ Then press `quit` button twice to exit to3d GUI
- Script version (no GUI): `to3d -prefix spgr I.*` would create a dataset with no user intervention
- Later: will give more complicated example of assembling data from 'naked' image files, where no header information is available

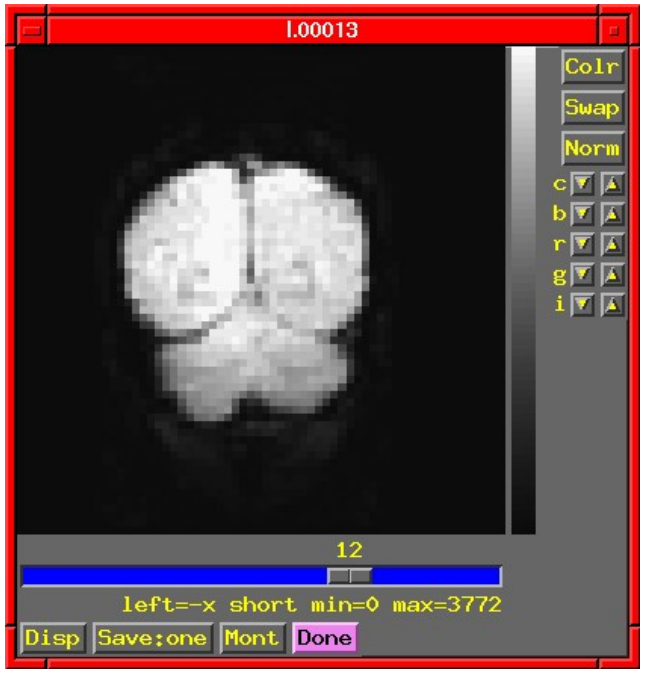
- Using to3d to assemble the EPI 3D+time dataset:
  - ⋆ `cd ../EPI_run1` — change directory, to get at images
  - ⋆ `ls` — to see what files are there (should see files `I.00001 ... I.02880`)
  - ⋆ We do not just do `to3d I.*` to create a 3D+time dataset
  - ⋆ For historical reasons, the time-axis information must be given on the to3d command line
    - ↪ Cannot be modified from GUI
  - ⋆ Command line: to3d -time:zt 18 160 0 alt+z I.*
  - ⋆ `-time:zt` means slices will be presented in order of space (z) then time (t)
    - ↪ This is the usual way slices are ordered, but -time:tz is needed at some sites
    - ↪ If in doubt, do `to3d I.*` or `aiv I.*`, use viewer to look at slices and see their order [`aiv`=AFNI Image Viewer program]
  - ⋆ `18 160` means that there will be 18 slices in z, 160 in t (2880 total)
  - ⋆ 0 means that the TR for volume acquisition will be read from the image headers
    - ↪ If not available, could put 2s instead of this 0
  - ⋆ `alt+z` means that the slices are gathered in alternating order in the +z direction
    - ↪ Most EPI acquisitions are really 2D multislice, spread out through time
    - ↪ AFNI header can contain information about slice timing offsets
    - ↪ Other possible modes: `zero` (for 3D), `@filename` (to specify each slice)


to3d
x orientation (across screen)
Right-to-Left
x voxel size (mm)
3.75
x origin (mm) [left edge]
118.125
R
y orientation (down screen)
Superior-to-Inferior
y voxel size (mm)
3.75
y origin (mm) [top edge]
118.125
S
z orientation (slices 0,1,..)
Posterior-to-Anterior
z voxel size (mm)
4
z origin (mm) [slice 0]
96.8
P
Datum: short [64x64]
cubical
square
irregular
Field of view (mm)
240
x axis centered
y axis centered
z axis centered
View:
Original View
TR = 2.000 (s)
NR = 160 Nz = 18
Copy geometry of this dataset
Anatomy parent is this dataset
Type of data in the images
3DIM_HEAD_ANAT
Type of anatomy in the images
Echo Planar
Byte Swap[2]
button help
View Images
Save Dataset
Field below not applicable
Session direct. for 3D Datasets
./
Prefix for 3D Dataset file
help
* Checking for *
* time series *
* outliers *
AFNI
PS->file
->printer
Done
Outlier Count (and Threshold)
Sub-brick Index
help <2>
to3d WARNING:
Significant outliers detected in these sub-bricks:
0 1 2 3 4 5 6 7 8 9 10 11
12 13 14 15 16 17 18 19 20 21 22 23
24 25 26 27 28 30 35 37 141 142 146 148
149 150 151 152 153 154 155 156 157 158 159
You should inspect the dataset for possible corruption.
[Outliers are defined as in program 3dToutcount.]
[Outliers early in an EPI time series may be due to]
[the longitudinal magnetization equilibration effect.]
[Other causes are subject movement, scanner problems,]
[or anything that makes a time series look irregular.]
[3dToutcount -save outnam dataset | 1dplot -stdin]
[can be used to make a dataset 'outnam' that marks]
[outlier voxels; see 3dToutcount -help for details.]

- ⋆ Outliers are data values that are very different from other values in the same time series
  - ↪ to3d reports sub-bricks (time points) that have a lot of outliers
  - ↪ You should use AFNI to look at these time points to see if there are major problems (e.g., head motion, scanner artifacts)
  - ↪ to3d `-skip_outliers` option lets you skip outlier detection step
  - ↪ Utility program `3dToutcount` can also report outliers and can even make a dataset with the 'outlier-ness' of each voxel value



★ Again, fields in GUI were filled in from data in the I.* headers



★ Coronal slices; will work with SPGR axials in AFNI

↪ Programs `3dresample` and `3daxialize` can rewrite datasets in new orientations

★ Note slice thickness and slice offset ("z origin")

↪ Values match experiment log (that's good)

★ Time information is displayed in GUI, but not editable

★ Have set "Type of anatomy" to "Echo Planar"

↪ Just acts as a reminder to user (not used elsewhere)

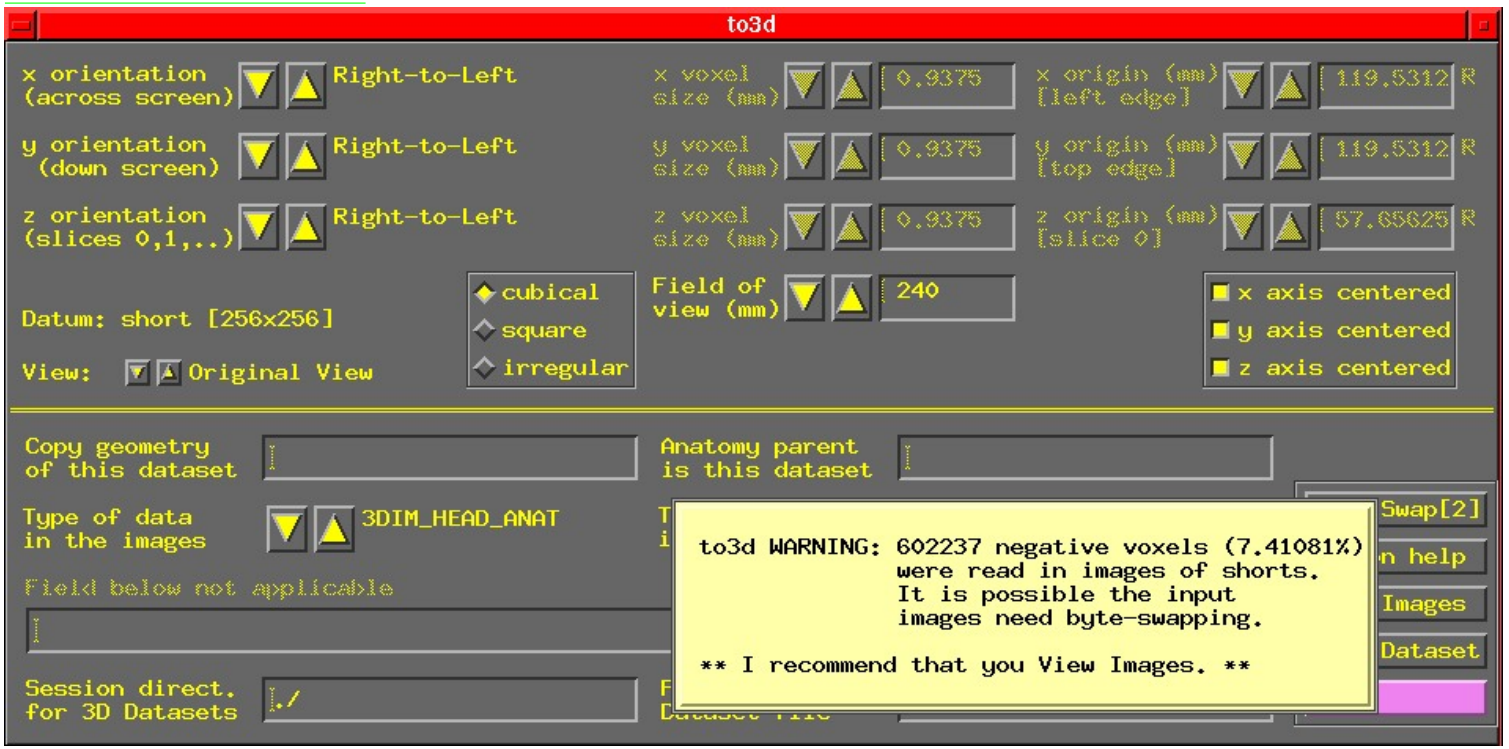
★ Script version: `to3d -time:zt 18 160 0 alt+z -prefix epi I.*`

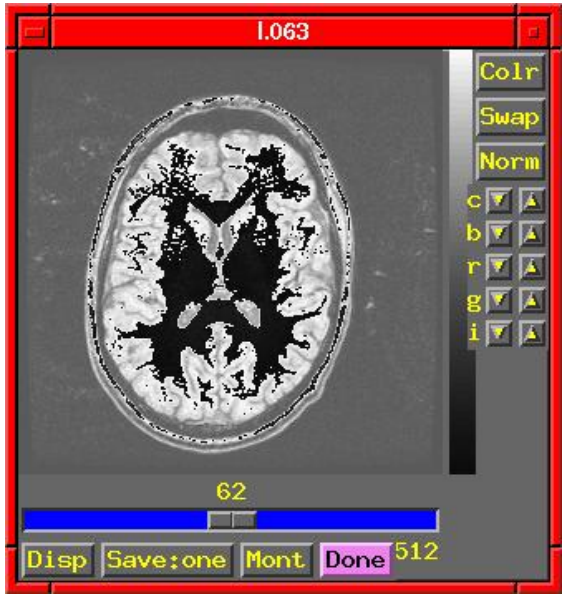
★ Program `3drefit` can be used to change some header items in an AFNI dataset after it is created

↪ Example: `3drefit -TR 1s epi+orig` will change the TR of the dataset to 1 second

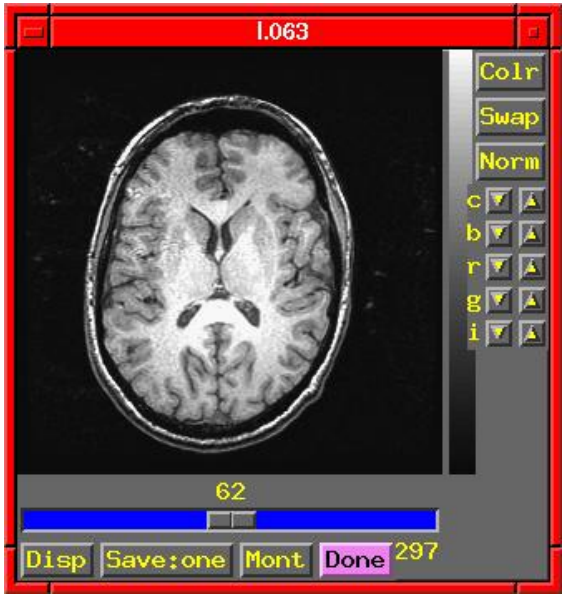
Assembling 'Naked' Images into AFNI Datasets

- 'Naked' image $\equiv$ image file without header data that AFNI understands
- User must supply geometrical information to to3d
  - ⋆ This is when the written experiment log is critical!
- The SPGR_naked directory contains the same SPGR images as before, but stripped of all header information
  - ⋆ Each file has 131072 bytes $= \mathbf{256 \times 256}$ 16-bit integers ('shorts')
  - ⋆ cd SPGR_naked, then to3d N.*





Linux/Intel computers

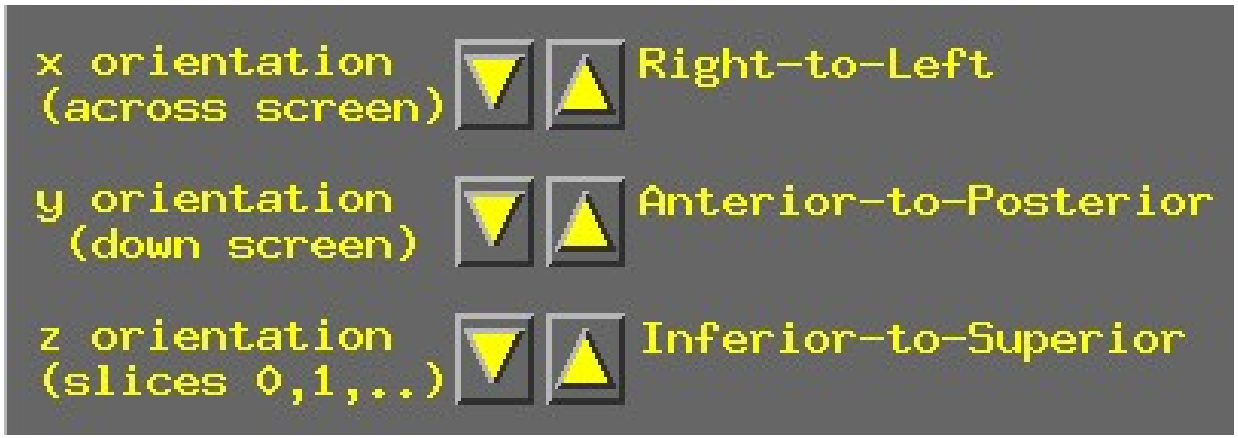


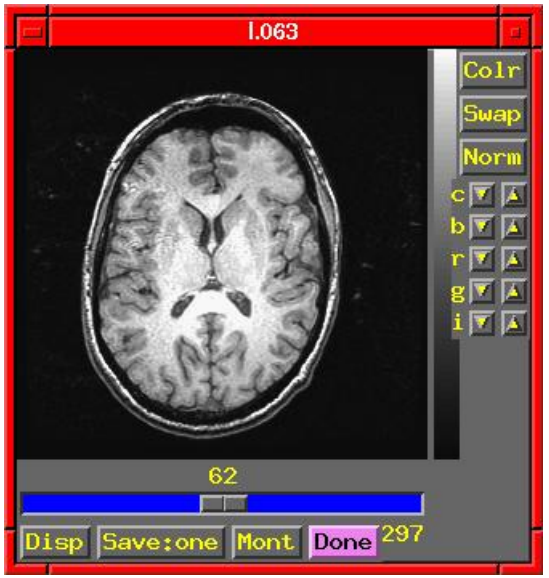
SGI/Sun/etc. computers

- On Linux/Intel computers: the peculiar appearance of the image shows that something is wrong:
  - ⋆ MR images from scanners are stored as shorts: 2 bytes per number
  - ⋆ Like a 2 digit decimal number: "93" means "$\mathbf{9 \times 10 + 3}$"
    - ↪ By universal custom, we write the "9" first
    - ↪ Could also write the same number as "39" (if we had a different custom)
  - ⋆ Customs for computers are not so universal
    - ↪ Sun and SGI systems store 2 byte numbers in reverse order from Intel
    - ↪ Result is that numbers are mangled (and some show up as negative)
    - ↪ Solution: press to3d's `Byte Swap[2]` button, and images are fixed!

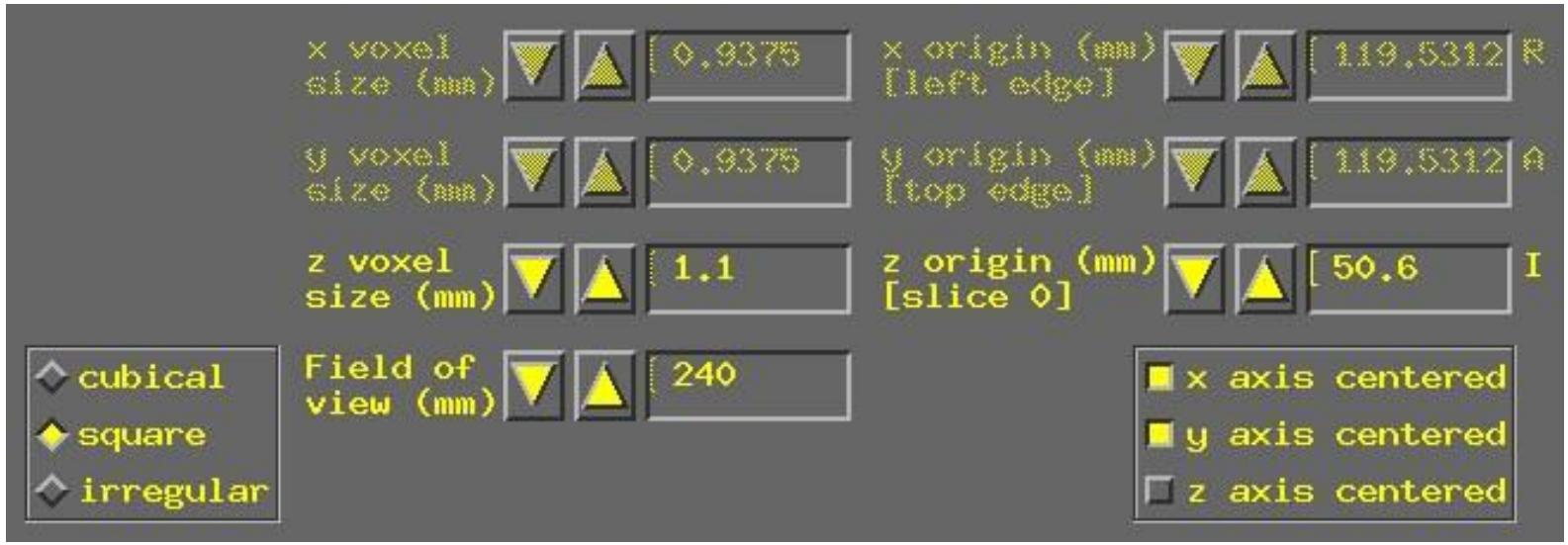
- Same to3d control panel (without the negative voxel warning):

- Above the double line: must fill out 3 types of geometry information:
  - ⋆ Left column: orientation of the dataset axes
  - ⋆ Middle column: size of the dataset images or voxels
  - ⋆ Right column: offset of the first slice





- Screen shot above shows correct orientation for this dataset
  - ⋆ Use the image viewing window to judge how images are laid out
  - ⋆ Click the arrows to scroll through the 6 possible options for each orientation to set the correct values
  - ⋆ "x orientation" of dataset is across the screen (left to right)
    - ↪ Must know subject's right from left
  - ⋆ "y orientation" of dataset is down the screen
  - ⋆ "z orientation" of dataset is in increasing slice index order
    - ↪ determine this by using the slider at the bottom of image window



- To set dataset geometrical size/location, experiment log sheet is essential
- Screen shot above shows setting slice thickness to 1.1 mm
  - ⋆ Default Field of view (FOV) of 240 mm is correct for these images
  - ⋆ Default voxel geometry of "cubical" is incorrect
  - ⋆ Must set geometry to "square" (x size = y size, z size different)
  - ⋆ Then set "z voxel size" to correct value (by typing in box)
- Screen shot shows setting center of first slice to 50.6 mm in Inferior (I) direction
  - ⋆ Default is that slices are centered in the magnet
  - ⋆ Probably not the case in the z direction
  - ⋆ Click "z axis centered" off
  - ⋆ Enter offset (here, 50.6 mm) into the "z origin" box

- Final required steps:
  - ⋆ Enter prefix for new dataset into "Prefix" text box at lower right of to3d control window
    - ↪ Choosing a good prefix is important for keeping datasets organized
  - ⋆ Press "Save Dataset" button
  - ⋆ Press "quit" (twice) to exit to3d
  - ⋆ The new dataset files should show up when you use command `ls`
  - ⋆ You might want to move them to some other directory
    - ↪ `mv *+orig.* ../afni` to move datasets to directory named `afni`, one level above
    - ↪ this directory was created when you unpacked `AFNI_sample_01.tgz`; it contains pre-made AFNI datasets from EPI and SPGR images

- Geometry parent lets you copy the geometry data from a pre-existing dataset and apply it to the dataset now under construction
  - ⋆ Enter name of pre-existing dataset into the `Copy geometry of this dataset` field
    - ↪ If in another directory, must include that in filename
  - ⋆ When you press 'Enter' or move the cursor from the text-entry field, to3d tries to read geometry parent dataset header
  - ⋆ If geometry parent has same spatial dimensions as current dataset, all geometry fields will be filled out
    - ↪ Does not affect the time fields, which must still be set using `-time:zt` or `-time:tz` on the command line
  - ⋆ Geometry parent very useful when constructing multiple EPI datasets from a single scanning session

- Using to3d in command line mode
  - ⋆ You can specify all needed inputs to to3d using command line options
    - ↪ For a full list of options, type `to3d -help`
  - ⋆ If enough information is present on command line to define a dataset, then the GUI will not be opened, and the dataset will be written to disk
    - ↪ If the command line is incomplete, then the GUI will be opened

★ For the SPGR dataset example:

```
to3d -xFOV 120R-L -yFOV 120A-P -zSLAB 50.6I-84.7S
     -prefix anat -2swap -spgr N.*
```

(this is all on one command line)

↪ `-xFOV 120R-L` says that the x axis of the images runs from 120 mm Right to 120 mm Left

↪ `-yFOV 120A-P` says that the y axis of the images runs from 120 mm Anterior to 120 mm Posterior

↪ `-zSLAB 50.6I-84.7S` says that the z axis of the slices runs from 50.6 mm Inferior to 84.7 mm Superior

- ▷ FOV means the distances apply from edge-to-edge of the images in that direction (x and y, in most cases)
- ▷ SLAB means that the distances apply to the centers of the outermost voxels (z=slice direction, in most cases)

↪ `-prefix anat` gives the prefix for output dataset filenames (in this case, `anat+orig.HEAD` and `anat+orig.BRIK`)

↪ `-2swap` means to byte-swap the images while reading them

↪ `-spgr` means to label this data as being of SPGR type

↪ `N.*`, as before, means to read the images from the files whose names start with the string "N." and end with anything ("*" is a wildcard)

- ⋆ For the EPI dataset example (if image files were 'naked'):

```
to3d -xFOV 120R-L -yFOV 120S-I -zSLAB 96.8P-28.8P
  -time:zt 18 160 2000 alt+z -prefix epirun1 -2swap -epan I.*
```

  (this is all on one command line)

  ↪ Options (with their arguments) can appear in any order

  ↪ Input image filenames always appear last

- Once you get used to it, command line usage for to3d is more useful than the GUI
  - ⋆ Usually need to create many datasets at once
  - ⋆ Can put commands in a script file and execute them
  - ⋆ Then edit that file to change a few things, and run it again
  - ⋆ Just create the file with your favorite Unix text editor (emacs, nedit, vi), typing each command on a separate line

    ↪ Long commands can be split across multiple lines by ending all but the last line with the "\" character

    ↪ There must not be a blank after the "\" !!!
  - ⋆ You can execute a script file by typing a command like `source filename`, which just means to read commands from "`filename`"
  - ⋆ As time goes on, you build up a set of scripts that automate various tasks for you, and ensure you do things the same way each time