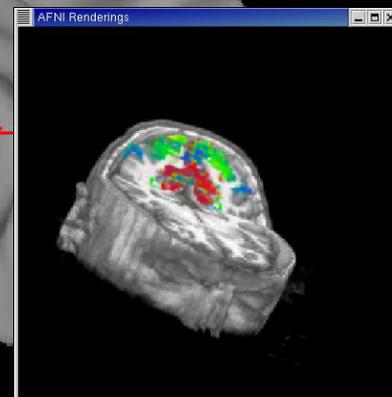
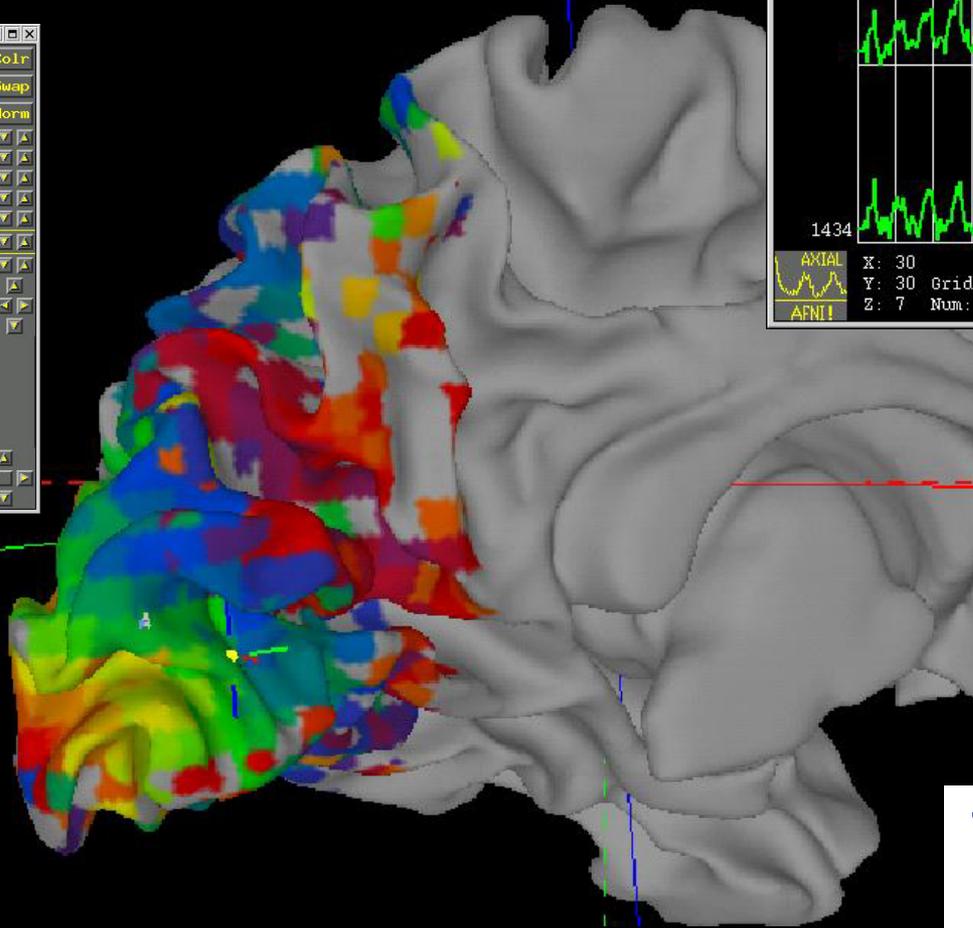
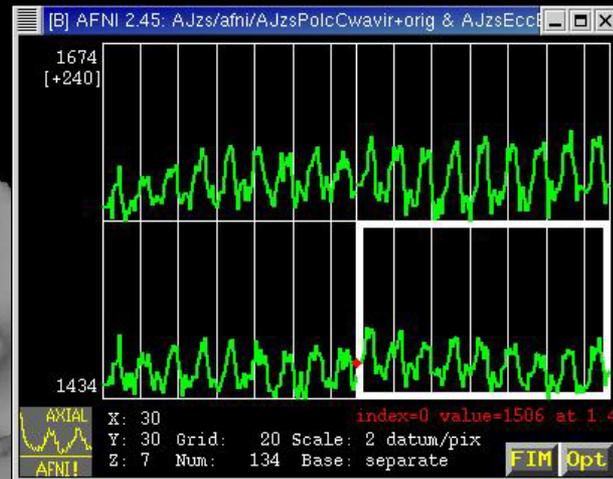
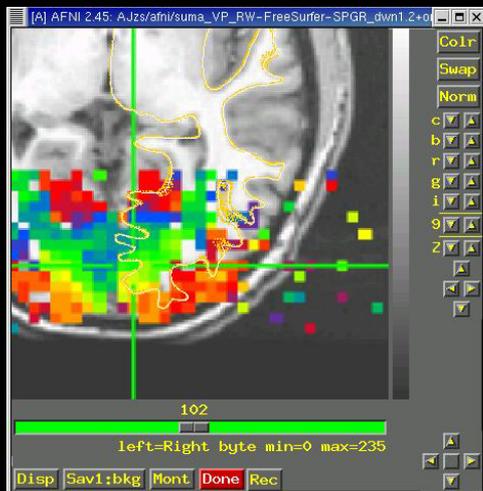
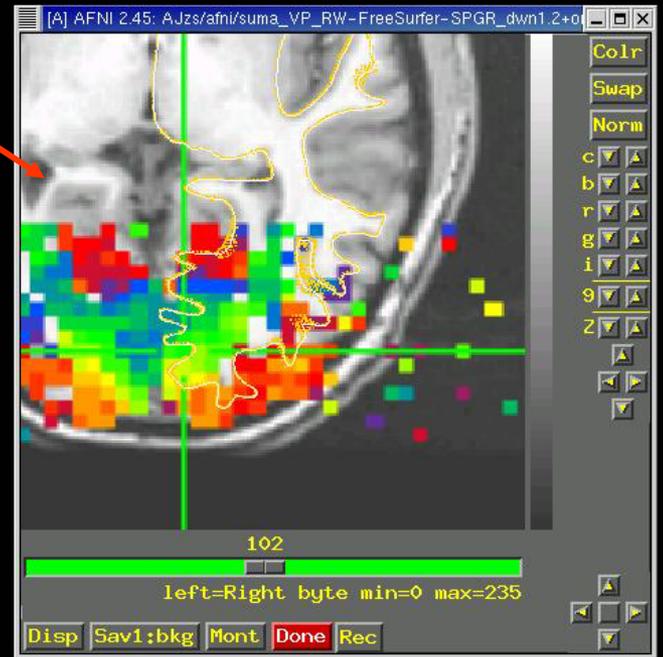
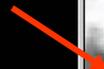


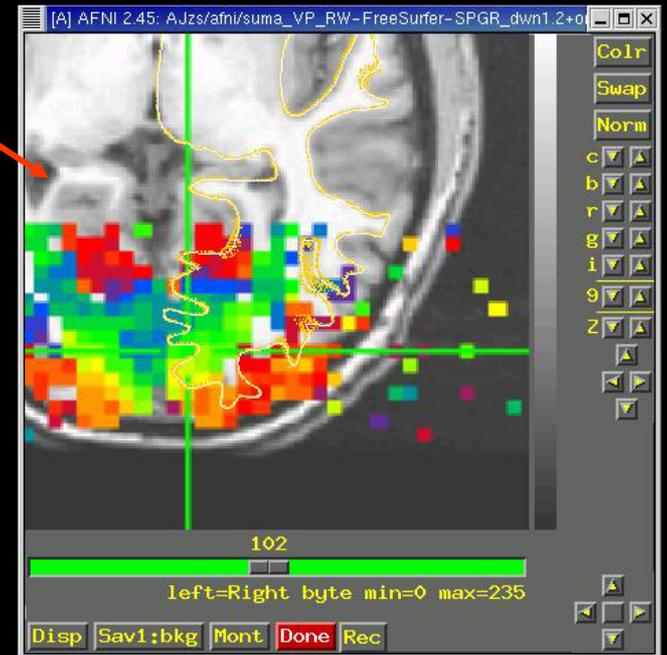
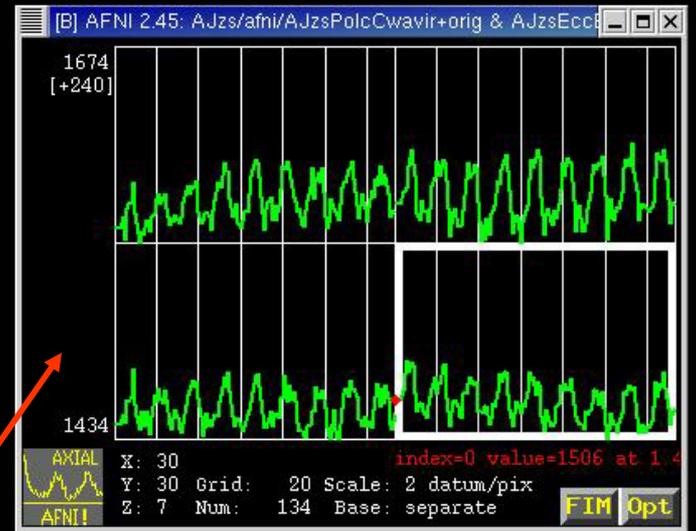
SUMA

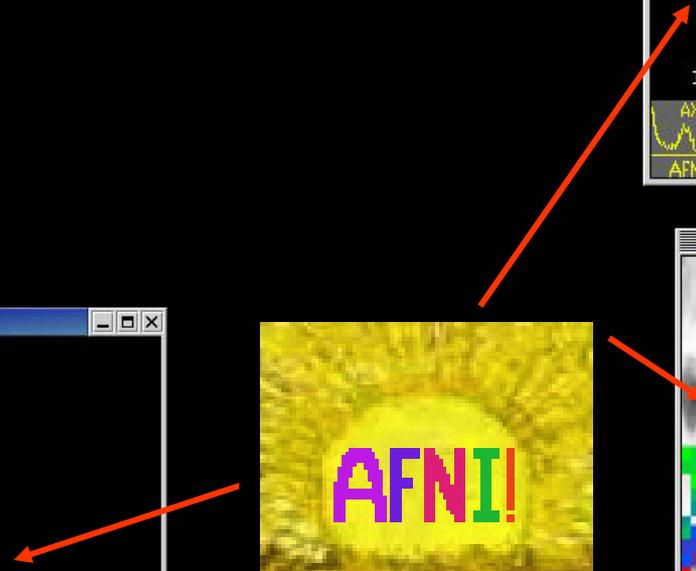
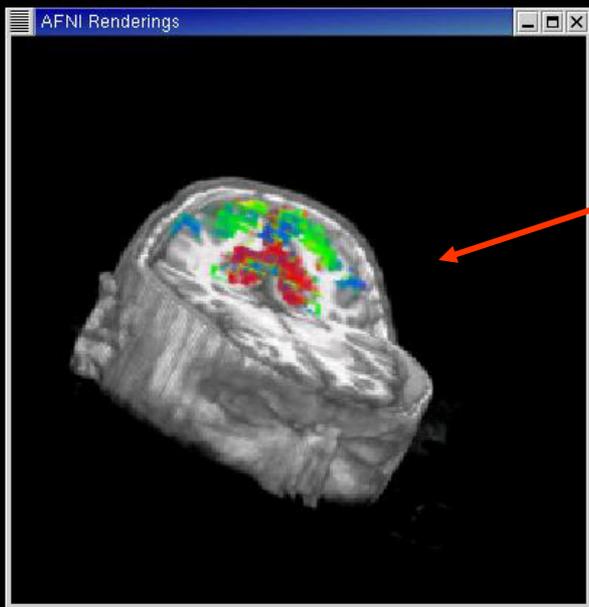
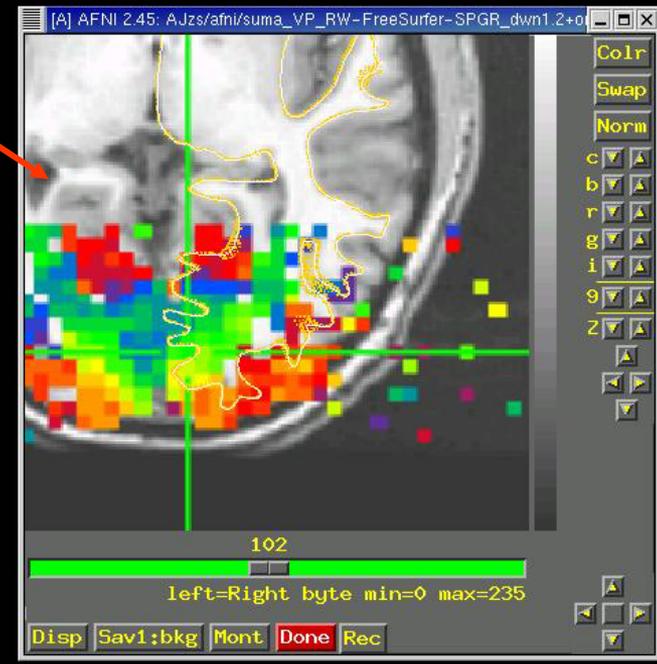
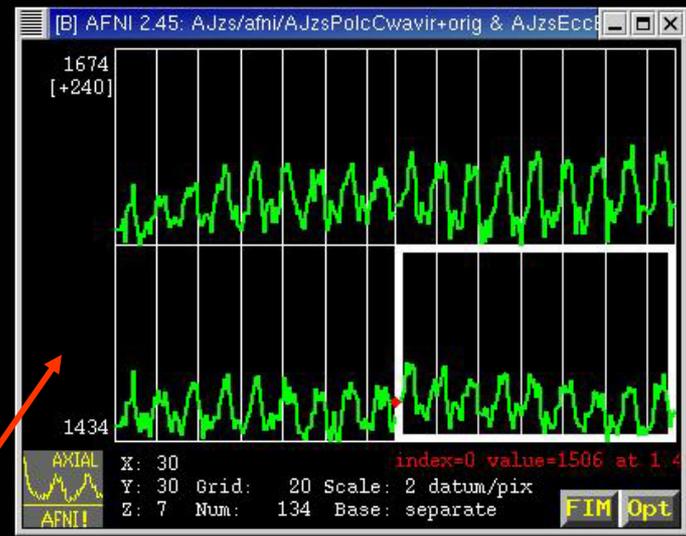


Ziad S. Saad & Robert W. Cox
Statistical & Scientific Computing Core

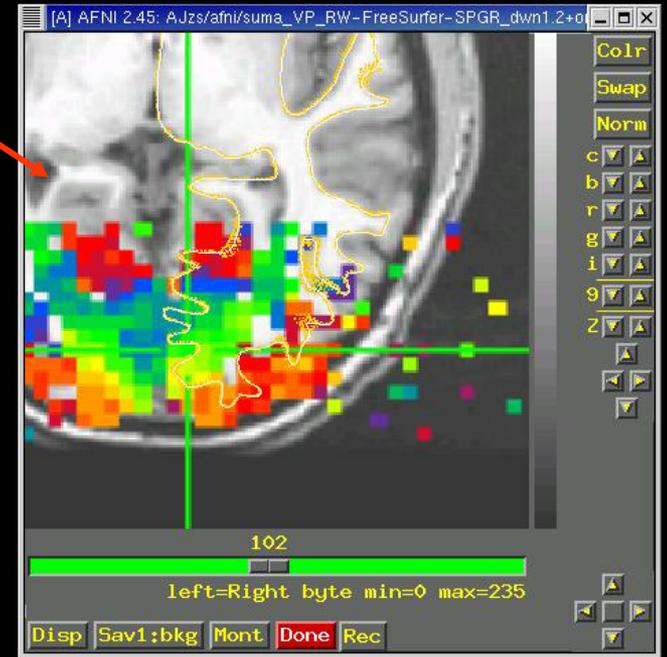
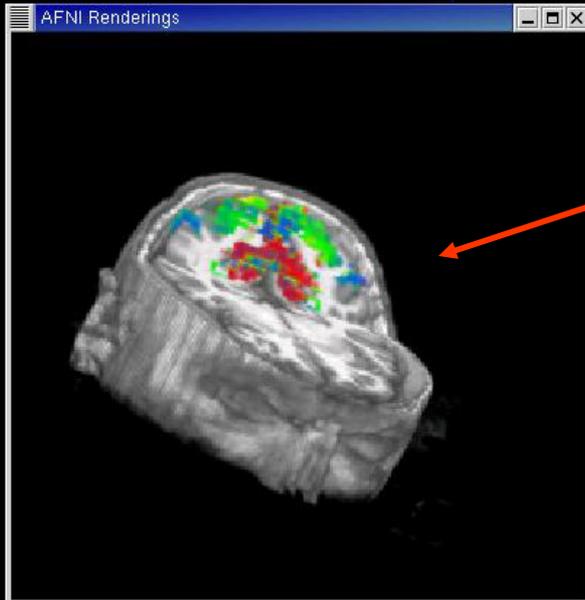
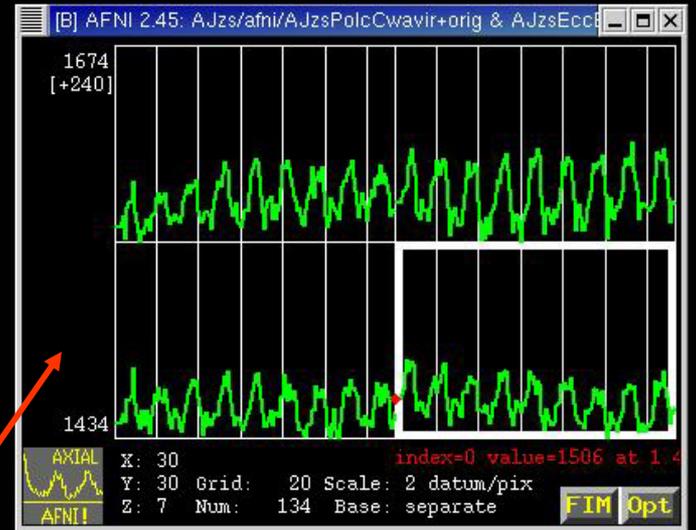
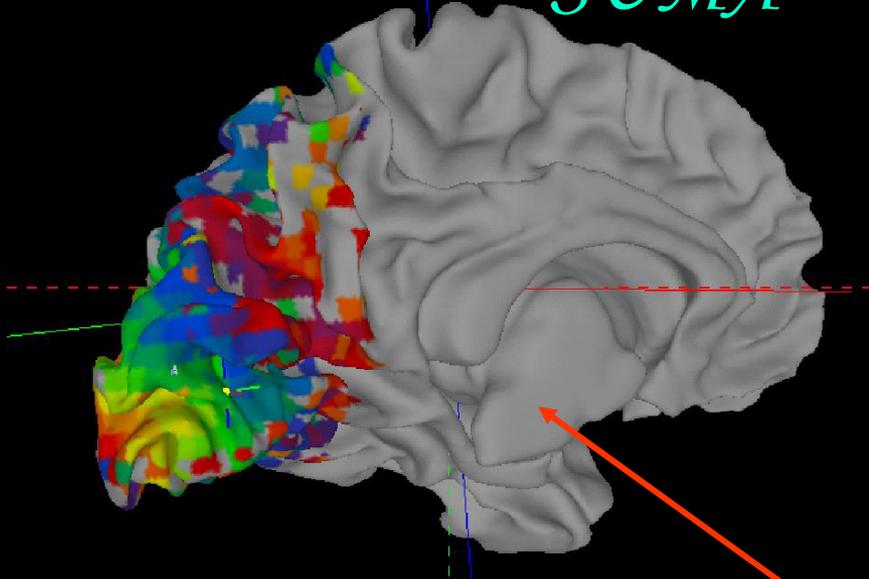








SUMA



SURface MAPPING with AFNI

- Surface mapping & viewing program tightly linked to AFNI
- Complements AFNI's slice and volume rendering modes
- Provides a framework for fast and user-customizable surface based analysis
- Supports surface models created by:
 - FreeSurfer** <http://surfer.nmr.mgh.harvard.edu>
 - SureFit/Caret** <http://stp.wustl.edu/resources/display.html>
 - BrainVoyager** <http://www.brainvoyager.com>
- Allows representation of sparsely defined 3D data

System Requirements

MESA Library 4.0.1 (an API closely resembling OpenGL)

<http://www.mesa3d.org/>

- Currently Running On:
 - Pentium IV, 1.9 Ghz / Linux
 - 1Gb RAM
 - nVIDIA GeForce4 graphics card 64 Mb
- Also Runs On:
 - SGI
 - SUN with openGL 1.2 or newer
 - Mac OSX with Xfree86, Motif and Mesa

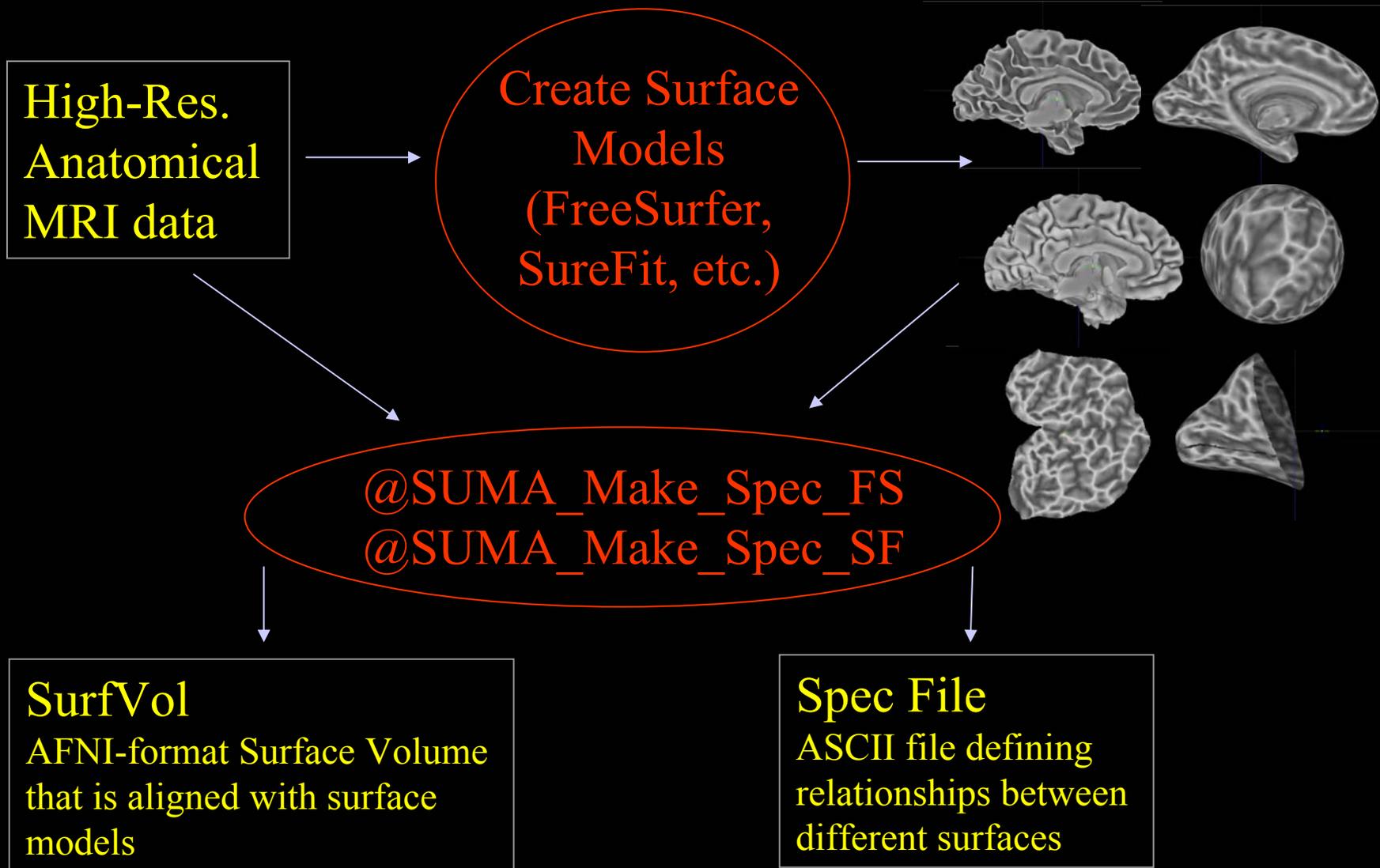
Installation

- Binaries for SUMA are available from:
http://afni.nimh.nih.gov/ssc/ziad/SUMA/SUMA_DownloadTable.htm
- On some machines (i.e. Linux), SUMA should be compiled locally for optimal performance.
- To compile SUMA you need to have:
 - Motif (both libraries and header files)
 - Mesa (OpenGL)
 - AFNI's source code distribution
- Detailed installation info for various systems:
http://afni.nimh.nih.gov/ssc/ziad/SUMA/SUMA_Installation.htm

Where do Surface Models Come From ?

- For surface based analysis:
 - Must create surfaces for individual subjects
 - PreSUMA:
 - Collect, align and average high-quality, high-resolution anatomical data
 - On NIMH's 3T-1, 4 MPRAGE data sets will do
 - Correct image non-uniformity
 - Using AFNI's 3dUniformize or the N3 normalization tool [J.G. Sled et al. 98]
 - Create and correct surfaces
 - Using FreeSurfer, SureFit or BrainVoyager
 - CircumSUMA:
 - Align surface with experimental data
 - Using @SUMA_AlignToExperiment
 - Map experimental volumetric data to surface
 - Using AFNI and SUMA
 - PostSUMA:
 - Fame, Fortune and Fortitude for some, if not all, or none of you
- For display (mostly) of Talairach data:
 - Use the Talairach surfaces created from the N27 brain data set using FreeSurfer.
 - Ready to use, no surface creation or alignment needed.

A: Preparing Surface Models for SUMA



A: Preparing Surface Models for SUMA

- Create the Surface Volume
 - SurfVol is an AFNI data set created from data used to create the surfaces.
- Create the surface specifications (spec) file
 - The Spec file defines the relationships between the different surfaces.
- Both the surface volume and the Spec file are created automatically
 - Using @SUMA_Make_Spec_FS for FreeSurfer surfaces
 - Using @SUMA_Make_Spec_SF for SureFit surfaces
- Demo for FreeSurfer surfaces:
 - cd suma_demo/SurfData
 - This is where FreeSurfer's directories reside
 - @SUMA_Make_Spec_FS -sid *DemoSubj*
 - Creates the SurfVol from the .COR files
 - DemoSubj_SurfVol+orig
 - Creates ASCII versions of surfaces found in the surf directory
 - lh*.asc and rh*.asc (if rh surfaces are provided)
 - Creates the Spec files for the left and right hemisphere surfaces
 - DemoSubj_lh.spec and DemoSubj_rh.spec (if rh surfaces are provided)
 - cd SUMA
 - afni -niml &
 - launches AFNI to allow the viewing of DemoSubj_lh.spec
 - the -niml option tells AFNI to listen to connections from SUMA
 - suma -spec DemoSubj_lh.spec -sv DemoSubj_SurfVol+orig

Check For Proper Alignment and Defects

- With both SUMA and AFNI running
 - Press ‘t’ in the suma window to establish a connection to AFNI and send the mapping reference surface(s).
 - If you open the surface volume in AFNI, you should see the surface overlaid on top of it.
 - **Switch Anatomy** to DemoSubj_SurfVol+orig and open an axial view. You will see a trace of the intersection of the surface with the anatomical slices displayed. You could also see boxes representing the nodes that are within +/- 1/2slice from the center of the slice in view. Colors and node box visibility can be changed to suit your fancy from the **Control Surface** button in AFNI.
 - Navigate through the volume in AFNI.
 - make sure you have an excellent alignment between volume and surface
 - make sure surface adequately represents areas of the brain that are difficult to segment
 - occipital cortex
 - inferior frontal and inferior temporal regions

Note: Viewed without the volume underlay, it is extremely difficult to tell if surface models with no topological defects accurately represent the cortical surface.
- The Surface Volume and the surfaces must be in perfect alignment.
 - If you have an improper alignment, it should be addressed here.
 - This should not happen for FreeSurfer and SureFit surfaces created in the standard fashion.
 - Watch for error messages and warnings that come up in the shell as the surfaces are read in. These messages should be screened once since they do not change unless the surface’s geometry or topology is changed.

Basic SUMA viewer functions

- Rotating the surface:
 - **Mouse button-1:** keep it down while moving the mouse left to right. This rotates the surface about the screen's Y-axis (dotted green). Let go of button-1.
 - Repeat with up and down motion for rotation about X-axis and motion in various directions for rotations mimicking those of a trackball interface.
 - Also try up/down/left/right arrow keys.
- Translating the surface:
 - **Mouse button-2:** keep it down while moving the mouse to translate surface along screen X and Y axes or any combinations of the two.
 - Also try shift+arrow keys.
- Zooming in/out:
 - **Both buttons 1&2 or Shift + button 2:** while pressing buttons, move mouse down or up to zoom in and out, respectively.
 - Also try keyboard buttons 'Z' and 'z' for zooming in and out, respectively.

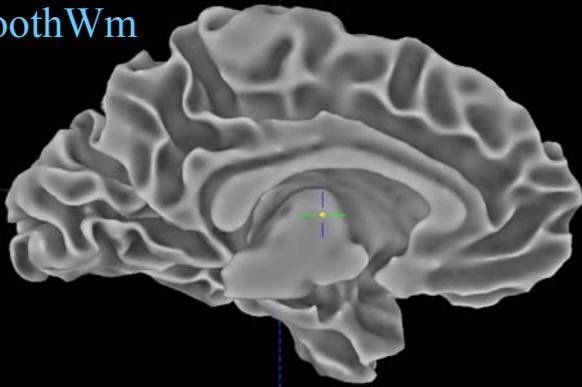
Basic SUMA viewer functions

- Cardinal views:
 - ctrl + Left/Right: Views along LR axis
 - ctrl + Up/Down: Views along SI axis
 - ctrl + shift + Up/down: Views along AP axis
- Picking a Node or Facet:
 - **Mouse button 3**: press over a location on a surface to pick the closest facet and node to the location of the pointer.
 - The closest node is highlighted with a blue sphere
 - The closest facet is highlighted with a gray triangle
 - Note the information written to the shell regarding the properties of the picked Node and Facet.
 - When connected to AFNI (after having pressed ‘t’), watch the AFNI crosshair jump to the corresponding location in the volume.
 - Conversely, position the crosshair in AFNI (left click) at a position close to the surface and watch the crosshair relocate in SUMA.
- Resetting the view point:
 - Press Home to get back to the original vantage point.
- Using momentum feature:
 - Press ‘m’ to toggle momentum on. Click the left mouse button and release the button as you are dragging the mouse.

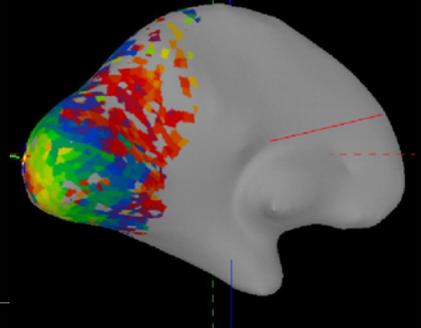
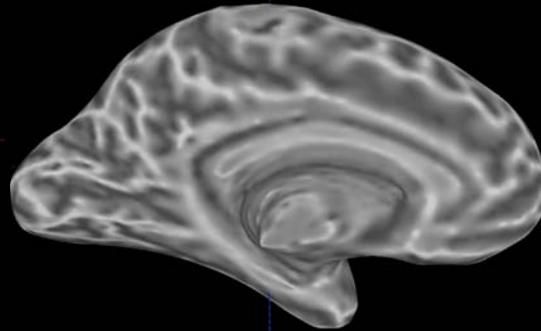
Relationships between surface models

- Surface Geometry:
 - refers to the spatial coordinates of the nodes forming the surface model
- Surface Topology:
 - refers to the connectivity between nodes forming the surface model
- Models with different geometry but similar topology are created for each surface model.
 - smooth white/grey matter
 - pial
 - inflated
 - spherical
 - flattened ...
- Some models' geometries are anatomically correct
 - Pial and/or white matter surfaces can be used for relating to volume data.
 - Inflated, flattened and spherical cannot be directly linked to volume data. The link is done via their corresponding anatomically-correct surfaces.

SmoothWm



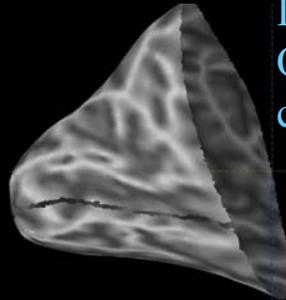
Inflated



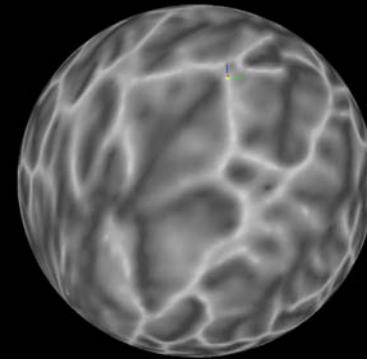
Pial



Inflated,
Occipital
cut



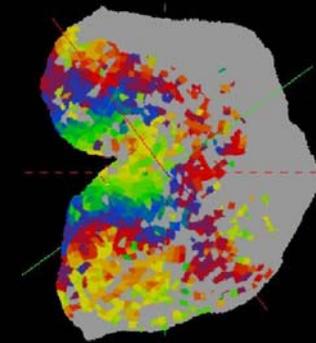
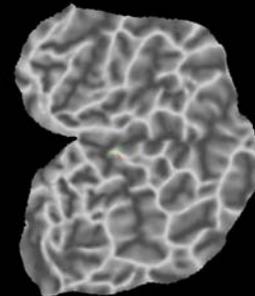
Spherical



Overlay of anatomically correct Pial and SmoothWm surfaces over anatomical volume



Flattened,
Occipital
cut



Sample Spec File

delimits comments

define the group
Group = DemoSubj

define various States
StateDef = smoothwm
StateDef = pial
StateDef = inflated

NewSurface
SurfaceFormat = ASCII
SurfaceType = FreeSurfer
FreeSurferSurface = lh.smoothwm.asc
MappingRef = SAME
SurfaceState = smoothwm
EmbedDimension = 3

NewSurface
SurfaceFormat = ASCII
SurfaceType = FreeSurfer
FreeSurferSurface = lh.pial.asc
MappingRef = lh.smoothwm.asc
SurfaceState = pial
EmbedDimension = 3

NewSurface
SurfaceFormat = ASCII
SurfaceType = FreeSurfer
FreeSurferSurface = lh.inflated.asc
MappingRef = lh.smoothwm.asc
SurfaceState = inflated
EmbedDimension = 3

NewSurface
SurfaceFormat = ASCII
SurfaceType = FreeSurfer
FreeSurferSurface = lh.sphere.asc
MappingRef = lh.smoothwm.asc
SurfaceState = sphere
EmbedDimension = 3

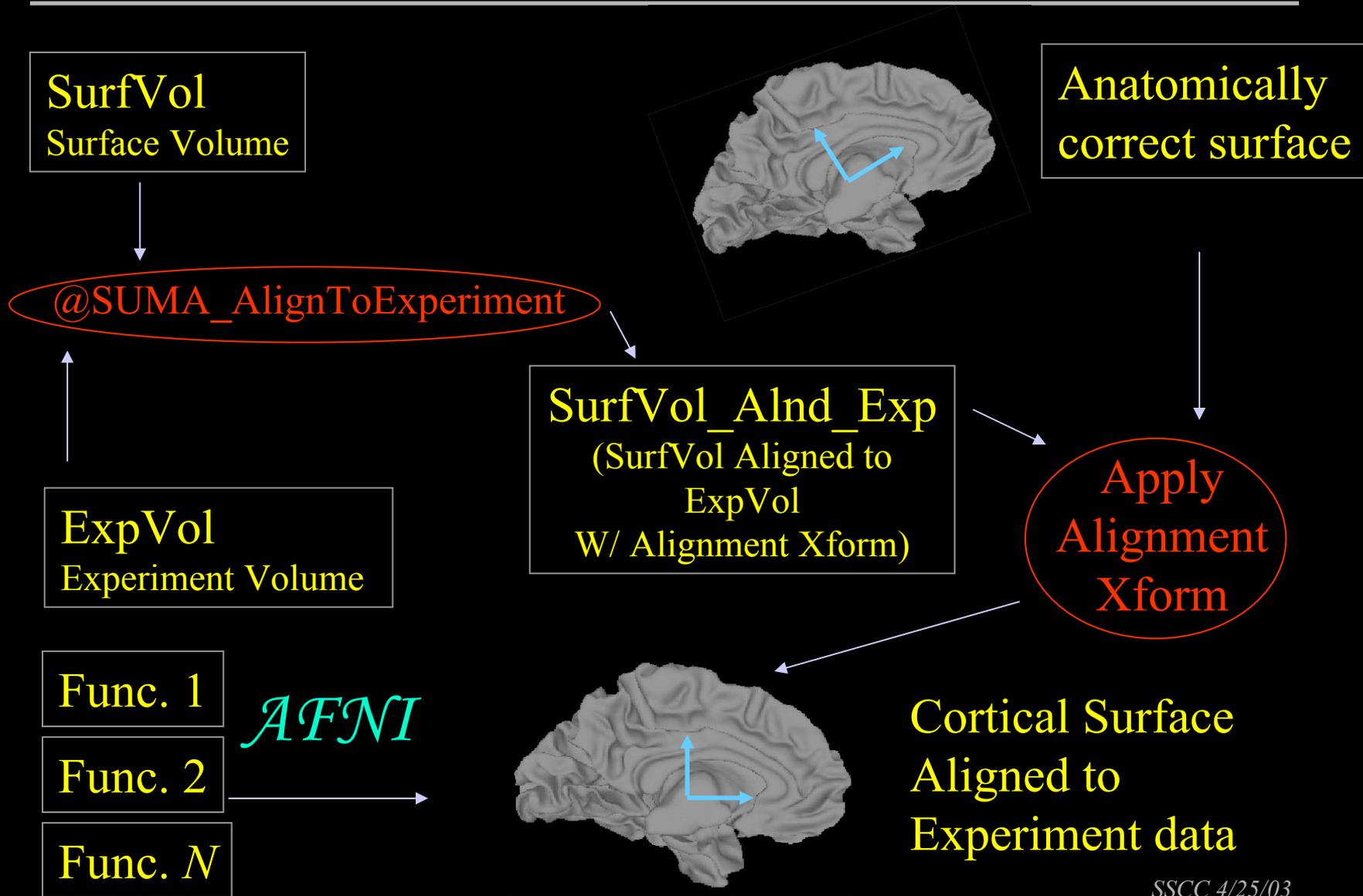
Details of the Spec file:

- The Spec file contains information about the surfaces that will be viewed. Information is specified in the format: *field = value*. The = sign must be preceded and followed by a space character. # delimit comment lines, empty lines and tabs are ignored. In addition to fields, there is also the **NewSurface** tag which is used to announce a new surface. Unrecognized text will cause the program parsing Spec file to complain and exit.
- The fields are:
 - **Group**: Usually the Subject's ID. In the current SUMA version, you can only have one group per spec file. All surfaces read by SUMA must belong to a group.
 - **FreeSurferSurface**: Name of the FreeSurfer surface. Do not include a path.
 - **SurfaceFormat**: ASCII or BINARY. Only ASCII is supported at the moment.
 - **SurfaceType**: FreeSurfer or SureFit
 - **SurfaceState**: Surfaces can be in different states such as inflated, flattened, etc. The label of a state is arbitrary and can be defined by the user. The set of available states must be defined with StateDef at the beginning of the Spec file.
 - **StateDef**: Used to define the various states. This must be placed before any of the surfaces are specified. Surface States that are not defined will cause the program parsing the Spec file to complain and quit.
 - **MappingRef**: Name of a surface to be used for mapping functional activity in the volume onto the cortical surface. This is necessary because it makes no sense to map volumetric data directly onto surfaces in certain states. The default for FreeSurfer surfaces is the smoothed gray matter/white matter boundary. For SureFit it is the fiducial surface. Use SAME when the Mapping Reference for a surface is the surface itself.
 - **EmbedDimension**: Embedding Dimension of the surface, 2 for surfaces in the flattened state, 3 for other.

Viewing the group of surfaces

- Switch Viewing States:
 - '.' switches to the next viewing state (pial then inflated etc.)
 - ',' switches to the previous viewing state
 - Navigate on any of the surfaces and watch AFNI's crosshair track surface
 - SPACE toggles between current state and Mapping Reference state
- Viewing multiple states concurrently:
 - 'ctrl+n' opens a new SUMA controller (up to 6 allowed, more possible but ridiculous)
 - switch states in any of the viewers
 - all viewers are still connected to AFNI
- Controlling link between viewers:
 - Open SUMA controller with 'ctrl+u' or View->SUMA Controller
 - SUMA controller crosshair locking options:
 - '-': no locking
 - 'i': node index locking (i.e. topology based)
 - 'c': node coordinate locking (i.e. geometry based)
 - SUMA controller view point locking
 - 'v': depress toggle button to link view point across viewers.
 - Surface rotation and translation in one viewer is reflected in all linked viewers

B: Aligning Surface w/ Experiment Data



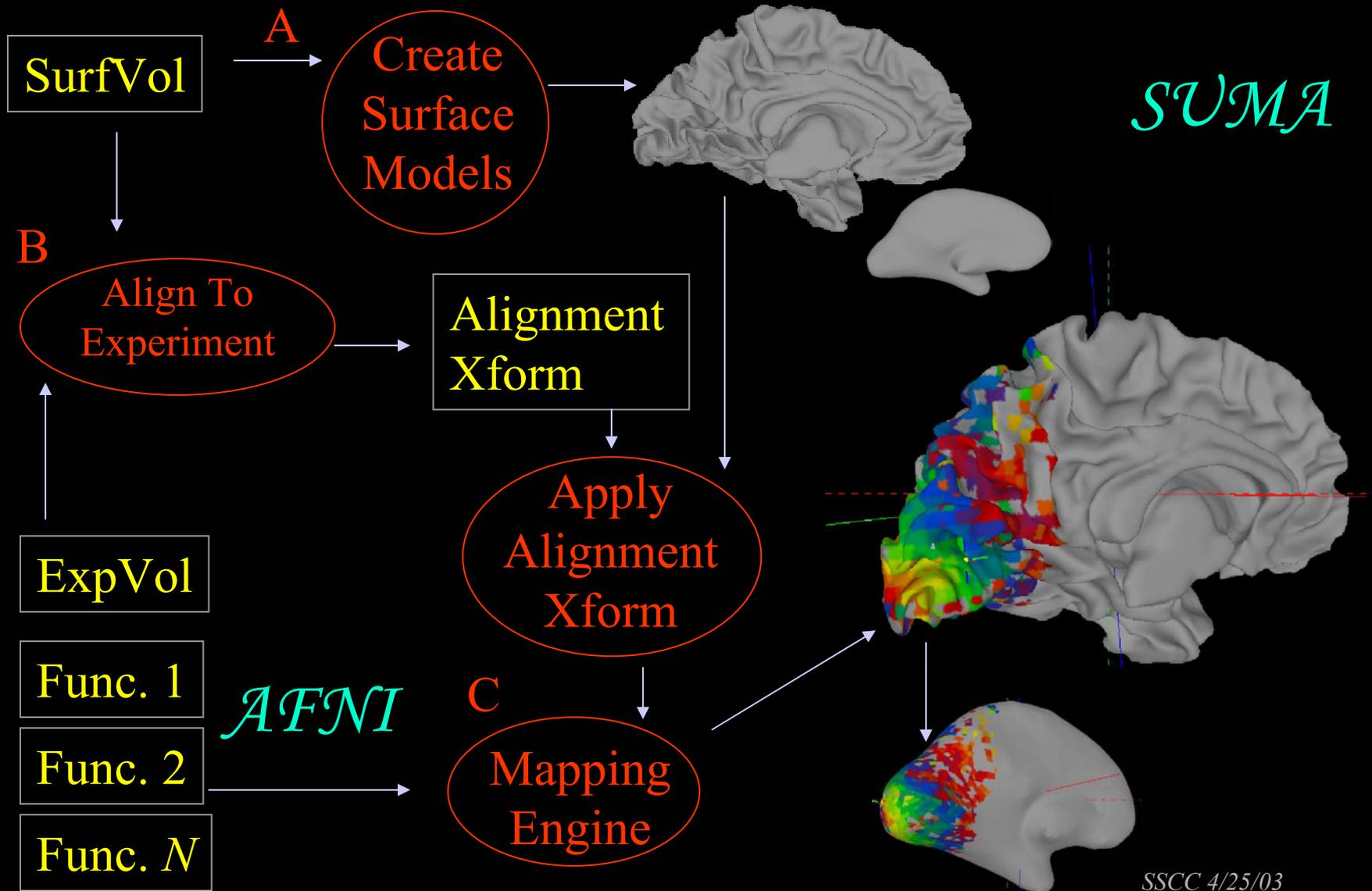
B: Aligning Surface w/ Experiment Data

- Surface Volume is aligned to experiment's anatomical volume with 3dvolreg
Brain coverage and image types should be comparable, not necessarily identical
- Functional data are assumed to be in register with experiment's anatomical
- Functional data are not interpolated
- Demo: (close previous SUMA and AFNI sessions)
 - cd suma_demo/afni
 - DemoSubj_spgrax+orig (the experiment's high-res. anatomical scan)
 - DemoSubj_EccExpavir+orig & DemoSubj_EccExpavir.DEL+orig (EPI anat. & func.)
 - @SUMA_AlignToExperiment DemoSubj_spgrsa+orig
../SurfData/SUMA/DemoSubj_SurfVol+orig
 - This script will use 3dvolreg to align the experiment's anatomical volume to the Surface Volume.
 - The script will take care of resampling (with 3dresample) the experiment's anatomical volume to match the Surface Volume if need be.
 - The output volume is named with the prefix of the Surface Volume with the suffix _Alnd_Exp (read Aligned to Experiment).
 - afni -niml &
 - We're launching AFNI to make sure that SUMA/DemoSubj_SurfVol_Alnd_Exp+orig and DemoSubj_spgrsa+orig are well aligned.
 - Switch Anatomy to DemoSubj_SurfVol_Alnd_Exp+orig
 - Open New AFNI controller (B)
 - Switch Anatomy in B to DemoSubj_spgrsa+orig
 - Make sure controllers A and B are locked (XYZ Lock, i.e. NOT IJK Lock). Do this through Define Datamode → Lock menu.
 - *continue next slide...*

B: Aligning Surface w/ Experiment Data

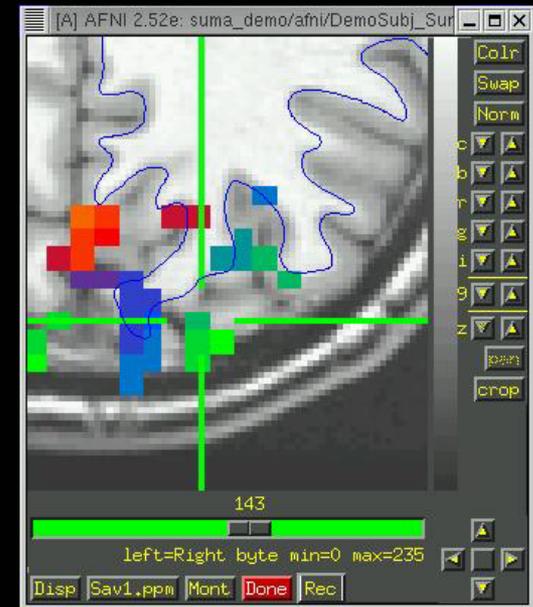
- Demo (continued)
 - Open the same views in both controllers (say Axial).
 - Click in one view and check if crosshair in other view points to a similar location.
 - Note how despite the difference in scan pulse sequence, SNR, and coverage, the alignment is pretty good.
 - average of 4 MPRAGE for surface model (~40 minutes to acquire at 3T)
 - one SPGR for experiment anatomy (~5 minutes to acquire at 3T)
 - Since DemoSubj_SurfVol_Alnd_Exp+orig is now aligned with the experiment's data and is of a superior quality, you should consider using it as your anatomical underlay.
 - Close controller B.
- `suma -spec ../SurfData/SUMA/DemoSubj_lh.spec -sv DemoSubj_SurfVol_Alnd_Exp+orig`
 - launching SUMA to make sure surface alignment is OK
- 't' to talk to AFNI
 - You should see a surface overlaid onto DemoSubj_SurfVol_Alnd_Exp+orig data set.
 - Alignment should be proper, otherwise you have a problem.

C: Mapping FMRI Data Onto Surface



C: Mapping FMRI Data Onto Surface

- Interactive mapping is done by AFNI
 - Mapping is done by intersecting Mapping Reference surface (the one sent to AFNI by SUMA) with the functional data volume.
 - Nodes inside a functional voxel receive that voxel's color
- Demo (continue from previous setup):
 - Switch Function to DemoSubj_EccExpavir.DEL
 - Define Function with:
 - Func: Delay
 - Thr: Corr. Coef.
 - Pos. color mapping
 - #20 color map
 - See Function
 - You should see the function on the surface model in SUMA.
 - The colors are applied to all topologically related surfaces
 - NOTE: Only AFNI controller A sends function back to SUMA
 - Change threshold in AFNI and watch change in SUMA
- The problem with intersection mapping
 - Only voxels intersecting the surface are mapped.
 - Can use Pial surface for Mapping Reference instead of SmoothWm.
- Other methods are in the works
 - mapping by intersection of cortical sheet (all of gray matter) with volume
 - interface for user-defined mapping routines
- Offline mapping of volume data to surface nodes
 - using 3dSurfMaskDump by **Rick R. Reynolds**



FMRI Data: Eccentricity Mapping

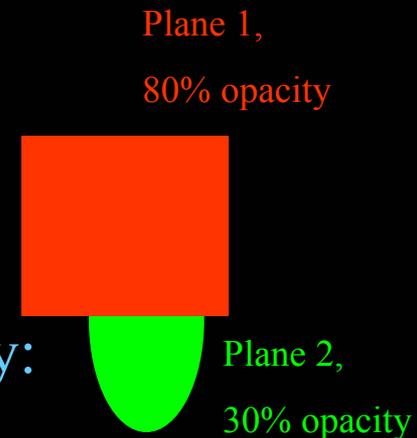
- Scan Parameters:
 - EPI: NIH-EPI, TR=2sec, 17 Coronal Slices, 134 samples, 3.75 x 3.75 x 4 mm
 - Anat: SPGR, 0.94 x 0.94 x 1.1, 120 axial slices
- Stimulus Timing:



- Activation delay estimated with 3ddelay
- Demo:
 - Rotate color map in AFNI and watch changes in SUMA
 - note how colors progress along the calcarine sulcus
 - try the dance on inflated and spherical surfaces

Color Overlay Planes

- Node color data are organized into layered planes
 - 2 commonly used planes are:
 - Surface Convexity (usually in gray scale)
 - AFNI Function (usually in color)
 - Planes are assigned to two groups
 - Background planes (like Convexity)
 - Foreground planes (like AFNI Function)
 - Many other planes can be added to either group.
- Mixing of color planes of the same group:
 - Planes are stacked based on their order and opacity.
 - Opacity of 1st plane does not affect color mixing.
- Node colors displayed on surface are obtained by:
 - 1st: mixing background planes
 - 2nd: mixing foreground planes
 - 3rd: layering foreground atop background planes
 - When foreground colors overlap background colors, they either mask the background or get attenuated by the latter's brightness.



Layering fore- & background planes

- Demo
 - View the surface in its inflated state with AFNI function
 - Turn foreground plane(s) off by pressing ‘f’ once
 - Now all you see is the background plane(s)
 - Turn background planes off by pressing ‘b’ once
 - Now all you see is the No Color color on all nodes
 - Turn foreground plane(s) back on with ‘f’
 - Now you have foreground without background
 - Turn background plane(s) back on with ‘b’
 - Now you have foreground atop background
 - Notice how you can still see the background underneath the foreground—this is due to the background brightness attenuation of the foreground colors.
 - Toggle background intensity attenuation off and on with ‘a’ and see the effect on the resultant maps.

Playing with color planes

- Demo (continue from inflated view with function)
 - ‘ctrl+s’ or View → Surface Controller to open the surface controller
 - Load in color plane frame to load a color plane file of your choosing
 - Color plane file format is: `i r g b` *per line*
 - choose `suma_demo/afni/lh.col`
 - Plane is placed atop of the foreground group
 - Its opacity is 1 so it will obscure the functional data
 - Note: Background attenuation is not affected by plane’s opacity.
 - try turning it on and off again with ‘a’
 - Now lower the opacity of `lh.col` and watch the function start to show through
 - You could put `lh.col` below the function
 - Switch Color Plane to get a list of available planes
 - Prefixes `fg:` and `bg:` denote plane’s group membership
 - Select `lh.col` and lower its order
 - Select `FuncAfni_0` and play with its opacity
 - Note: You can’t make a plane change its group membership, yet.
 - You can’t delete a loaded color plane yet, but you can hide it.

Offline mapping of volume data

- Volumetric data can be mapped to surface nodes using the program: `3dSurfMaskDump` by R. Reynolds.
- The mapping is done by intersection as in AFNI.
 - New mapping methods will be added to this program as they are developed.
- The output is an ASCII file. Each row contains the following:
 - node index
 - AFNI brick index (1D Voxel index)
 - AFNI IJK index (3D Voxel index)
 - each sub-brick value mapped to the node
- Very useful for performing node-based statistics
- Help and examples available:
 - `3dSurfMaskDump -help`

Talairach data display, for panache

- Without creating individualized surfaces, you can display data on a Talairach-ed surface model.
 - The surface models were created with FreeSurfer from the N27 data set.
(Holmes, CJ et al. JCAT 1998)
 - Anatomical volume and surface models were Talairach-ed using AFNI and SUMA, respectively.
(Surfaces models created by Brenna Argall)
- Demo: (close previous SUMA and AFNI sessions)
 - `cd suma_demo/afni_tlrc`
 - You should find Anat+tlrc and a functional data set in Talairach space
 - `afni -niml &`
 - setup function as indicated in previous slide
 - `suma -spec ../SurfData_tlrc/SUMA/lh.tlrc.spec -sv Anat+tlrc`
 - start communication with AFNI
 - map function onto surface
 - appreciate the limitation of this approach
 - ignore the limitation and make cool pictures

Saving your rendered image

- The rendered images can be saved in many ways (usually)
 - using ‘w’ in SUMA (except on Mac)
 - renders the scene to a pixmap and writes it out to an encapsulated postscript file (.eps)
 - eps file can be viewed/saved in other formats using image viewing programs such as gimp or xv
 - enlarge the rendering window to the maximum to get the best resolution
 - This option crashes the X server on certain Linux systems. This is not caused by SUMA. Sometimes upgrading the OS or the graphics card drivers solves the problem.
 - The program SUMA_pixmap2eps is a barebones program to test if rendering to a pixmap works on your machine. If successful, this program saves the rendering of a dinosaur to files dino.bw.eps and dino.rgb.eps
 - using screen capture programs
 - You can capture the rendering window with programs like gimp and xv, except on the Mac.
 - On the Mac you must use the Mac program Grab to capture the rendering window.
 - what you see in the window is what you get in the image so I recommend that you do NOT use the ‘w’ option, when possible
 - screen capture has been tested successfully on SGI/Linux/Mac/SUN

Auxiliary Programs

- **3dSurfMaskDump (by R. Reynolds)**
 - Maps volumetric data to surface nodes
- **MakeColorMap**
 - Creates colormaps for use with AFNI
- **ConvertSurface**
 - Converts surfaces between the different formats that SUMA can read:
 - FreeSurfer
 - SureFit
 - Simple ASCII matrix format
 - PLY format
 - Converts surface coordinates to Talairach space
- **ScaleToMap**
 - transforms a set of node values to node colors according to specified color map
- **In the works**
 - **CreateIcosahedron (by B. Argall)**
 - Creates Icosahedral meshes of varying node counts
 - **MapIcosahedron (by B. Argall)**
 - transforms the topology of surface models to standard models without distorting the geometry (requires FreeSurfer's spherical mapping).
 - **CompareSurfaces (by S. Japee)**
 - calculates the distance along the normal from one surface to the next
 - Something to turn surface ROIs to volume ROIs

Drawing Surface Based ROIs

- Demo
 - ‘Ctrl+d’ or Tools → Draw ROI to open ROI drawing tool
 - Drawing is done with the right (pick) mouse button clicked and the shift key depressed and lonely (*s&rc*).
 - When you *s&rc* for the first time, a new “drawn ROI” is created.
 - Note the Parent: field in the ROI frame is now filled
 - Move the mouse to a new location and *s&rc* again to draw a line from the location of the first click to a new one.
 - Or *s&rc* while dragging the mouse to create a smoother line.
 - Both drawing methods might fail if you are drawing over very rough terrain.
 - When that happens, you continue from where the drawing stopped.
 - Drawing in 3D looks easy but it isn’t.
 - Use Undo and Redo when you get your drawings messed up.
 - To close a loop you can press Join button or double *s&rc*
 - To fill a closed loop, *s&rc* inside the loop.
 - Note: If you have patterns that make a figure 8 jealous you might have to do multiple fills.
 - Press Finish when you are done drawing and have set the drawing label and value to your liking.
 - Now you can draw another ROI, and another and another.
 - Drawing can be started/continued on any of the related surfaces
 - When you hit join, the loop is closed based using the surface where the ROI was created
 - Switch ROI to switch between ROIs and delete/modify them.
 - Load to load ROIs from file
 - Save to save ROIs to file
 - ID/Niml Format of ROI file
 - This/All which ROI to save into the file
 - NOTE: ROIs are drawn with little spheres, so rendering becomes much slower if your ROI is very large. This will be modified in future versions.

Getting Helped

- `suma -help` for SUMA's command line usage
- 'ctrl+h' opens a window with help for SUMA's use
- BHelp should provide help for some buttons in the GUI interface
- SUMA's web site:
 - <http://afni.nimh.nih.gov/ssc/ziad/SUMA/>
 - http://afni.nimh.nih.gov/ssc/ziad/SUMA/SUMA_Installation.htm
 - http://afni.nimh.nih.gov/ssc/ziad/SUMA/SUMA_doc.htm
- AFNI's Message Board
- E-mail to ziad@nih.gov
- If you can't get help here, please get help somewhere.

AFNI ← communication → SUMA

- Programs are independent and communicate using NIML formatted data elements
 - via shared memory if both programs are on the same computer
 - via network sockets otherwise
- NIML: NeuroImaging Markup Language developed by Dr. R.W. Cox
 - NIML will be the main format for SUMA's data storage.
- NIML API library for packing/unpacking data is available and documented.
- Communication protocol allows any independent program to communicate with AFNI.
- Advantages include:
 - Programs execute on separate machines
 - Fast development
 - Screen real-estate
- Blemishes include:
 - Only one AFNI can be listening for connections
 - Only one SUMA can connect to AFNI

In the works

- Creating Volume ROIs from Surface ROIs
- Node based statistical computation such as 3dcalc, 3ddeconvolve, 3danova, etc...
- Surface based operations such as smoothing, clustering, labeling, gradients, contours
- Surface metrics such as curvature, distance, area
- New data mapping methods
- Enhance SUMA, Surface, and Viewer controllers
- Display surface based functional data
- Controls for manipulating color maps and color selection

Thanks to:

Developers:

Brenna Argall
Shruti Japee
Rick Reynolds



Critics:

Mike Beauchamp
Pat Bellgowan



Data and Documentation:
Peggy Christidis

Testing: Samia Saad

Getting Help Somewhere:
Richard Doucette

SysAdmin:
Alex Clark



MRI Pulse Sequence and Segmentation
Hauke Heekeren
Sean Marrett

