



# Programming Half-Week January 2006

## Clinical/End-User Applications

- Rule Based Segmentation Slicer Module
- Non-rigid intrasubject registration of T2 EPI to T1 conventional
- Shape analysis Visualization tool
- Client-Server based Time Series Navigation for fMRI Statistical Analysis

## Algorithms/Algorithm Infrastructure

- Joint Registration and Segmentation Framework
- Affine Invariant Anisotropic Smoothing ITK Filter
- Riemannian DTI Filters
- Automated image mosaicing and feature tracking for Electron Microscopy data
- Angular Smoothing and Interpolation of DW-MRI
- Tissue Classification from MRI with Neighborhood Statistics
- Volumetric Mesh Generation for Orthopaedic Applications (with UIowa)
- Integrating Tube Visualization for Tractography Generated in Slicer (with NWU)
- Block Matching Based Registration Algorithm in ITK (with ITC Canary Island)

## Software Infrastructure

- Slicer 3 Architecture
- Slicer 3 web presence
- Develop API for communication between command line executables and GUI's
- Slicer 3 Data Model
- Slicer 3 UI
- Slicer 3 and IGSTK integration (with Georgetown)
- Large Scale Job Submission with Slicer 3
- itku: Command-line ITK interface
- Graphical framework to construct/ execute complex scientific analyses of data
- Simple to use UNC shape analysis LONI pipeline

■ Contd from 2005 (7)

■ Collaborator (4)

■ New



# History: Programming Week June 2005

---

## Applications

- Quantitative Fiber Tract Analysis
- Rule-based DLPFC Segmentation
- Non-Rigid EPI Registration

## Algorithms

- Shape Description
- Shape Analysis
- Bayesian Segmentation
- 2d/3d point-landmark Detection
- Editing 3D Adaptive Tetrahedral Mesh Generation
- Flux Diffusion

## Software Infrastructure

- ITKu
- NRRD IO
- Slicer DTMRI Nightly Testing
- LONI Pipeline Integration of UNC Shape Analysis Pipeline
- Large Scale Algorithm Job Submissions via Condor
- Slicer 3.0 Architecture



# Slicer Rule-Based Segmentation Module:

Port existing rule-based Matlab code to a Slicer module.

## Team

Delphine Nain, GT (algorithms) (contact)  
 John Melonakos, GT (algorithms)  
 Shawn Lankton, GT (algorithms)

Alex Yarmakovich, Isomics (software)

Jim Fallon, UCI (clinical)  
 Martha Shenton, Harvard (clinical)

## Plan/Expected Challenges/Publication

Algorithms: port working Matlab rule-based segmentation code to a Slicer module.

Software: computational speed. Intuitive user interface.

Clinical: Validation of results.

This is an expansion of the previous programming week projects:  
 ITK Bayesian Filter & DLPFC Semi-Automatic Segmenter

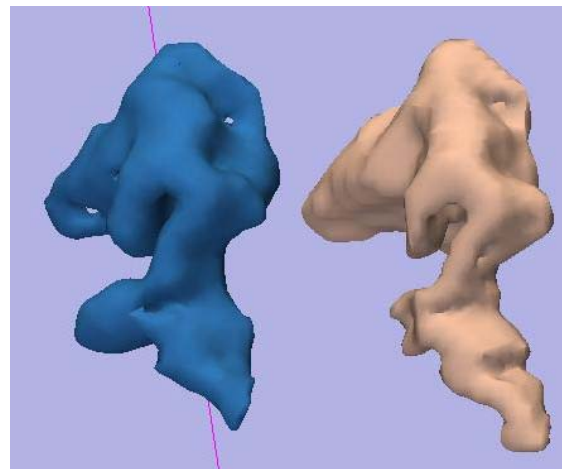
Expected Date for Insight Journal Publication: End of February 2006

## Accomplished by end of Programming Week

This project consists of 3 main pieces:

- 1) GUI – A preliminary GUI was developed
- 2) ITK Bayesian – This was wrapped in VTK and successfully compiled
- 3) VTK Thumb Extractor – A preliminary VTK filter has been developed

(Add picture to illustrate accomplishment)





# DLPFC Semi-automatic Segmentor:

With minimal user interaction, this rule-based algorithm will segment the DLPFC from an input volume. This is a Slicer project but may have components that are appropriate for ITK.

## Team

Ramsey Al-Hakim, GT (algorithms) (contact)

John Melonakos (algorithms)

Alex Yarmakovich, Isomics (software)

Jim Fallon, UCI (clinical)

## Plan/Expected Challenges

Algorithms: start with the Matlab implementation and port to Slicer then explore porting to ITK.

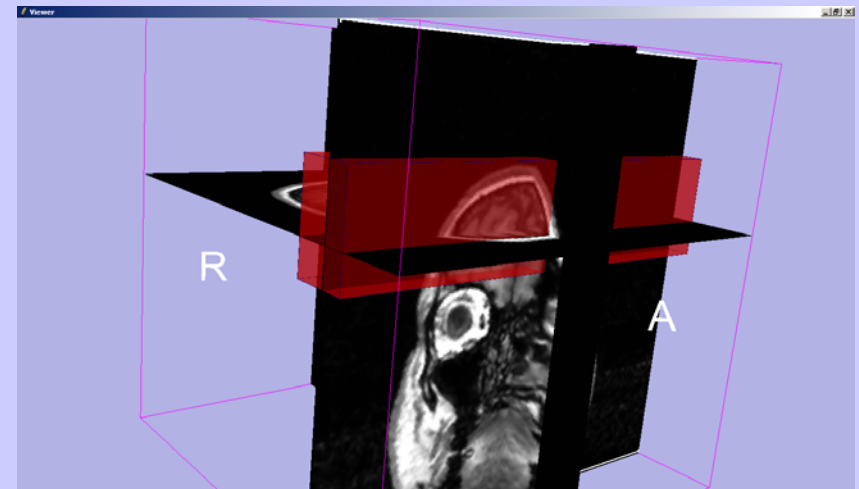
Software: computational speed. Intuitive user interface.

Clinical: Validation of results.

## Accomplished by end of Programming Week

Build Slicer user interface.

Final Step: Wrap ITK Bayesian Classifier Filter in VTK, and add to module.





# EPI to conventional T1 registration:

Develop software tool which can improve accuracy and/or robustness of affine registration of an EPI image to a conventional T1 image.

## Team

Josh Snyder, MGH (algorithms)  
Dennis Jen, MGH (algorithms)  
Alex Yarmarkovich, BWH

Xiaodong Tao, GE (engineering)  
Luis Ibanez, Kitware (engineering)

Andy Saykin, Dartmouth (clinical)

## Plan/Expected Challenges/Publication

1. Comparison tool to evaluate output quality compared to affine tools.
2. Working BSpline implementation.
3. Tuned BSpline implementation compared to affine implementations (Slicer, FSL's flirt).

Expected Date for Insight Journal Publication:

## Accomplished by end of Programming Week, 2006 (Add picture to illustrate accomplishment)

Working tool.  
Apparent better match than affine.  
Still tuning.





# EPI to MR Registration:

Develop software tool which can perform robust non-rigid, intrasubject registration between an EPI image and a conventional structural image.

---

## Team

Dave Tuch, MGH (algorithms) (contact)  
 Josh Snyder, MGH (algorithms)  
 Sandy Wells, BWH (algorithms consultant)

Luis Ibanez, Kitware (software)

Andy Saykin, Dartmouth (clinical)

## Plan/Expected Challenges

Algorithms:

- Write testing harness for validation of results.
- Identify suitable registration method available in ITK and optimal parameters.
- Refine as necessary

Software: Provide guidance on optimal use of ITK reg methods.

Clinical: Validation of results.

## Accomplished by end of Programming Week, Boston 2005

Motivated fuller support for image orientation in Slicer and ITK  
 Expanded understanding of metric and optimizer parameters  
 Produced working rigid registration  
 Identified possible approaches for non-rigid component



# Shape Analysis Visualization: MeshVisu

Visualizing shape analysis results on surfaces

## Team

Ipek Oguz, UNC (algorithms) (contact)  
Martin Styner, UNC (algorithms)

Sebastien Barre, Kitware (software)

## Plan/Expected Challenges/Publication

First programming week: MeshVisu is a tool for visualizing 1D attributes on a surface, such as p-value maps, etc.

Algorithms: Extend MeshVisu's functionality to 3D attribute sets (vector fields, ellipsoid fields on surfaces).

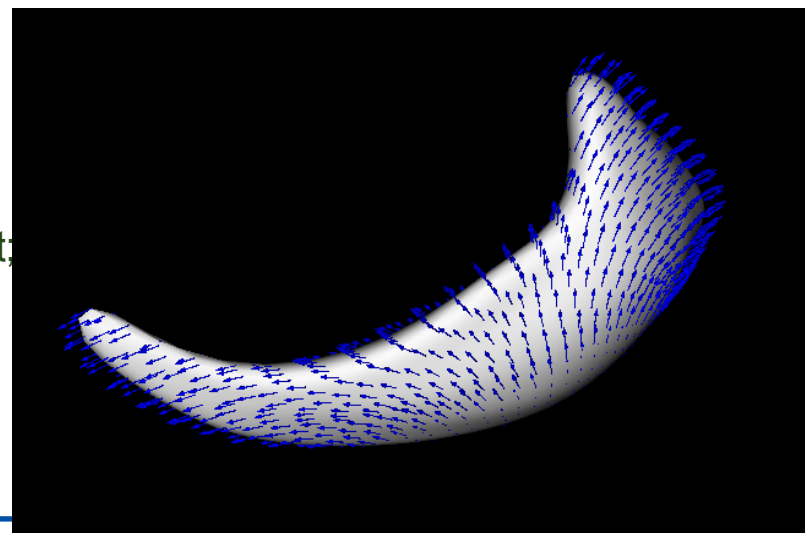
Software: Porting the user interface to KWWidgets.

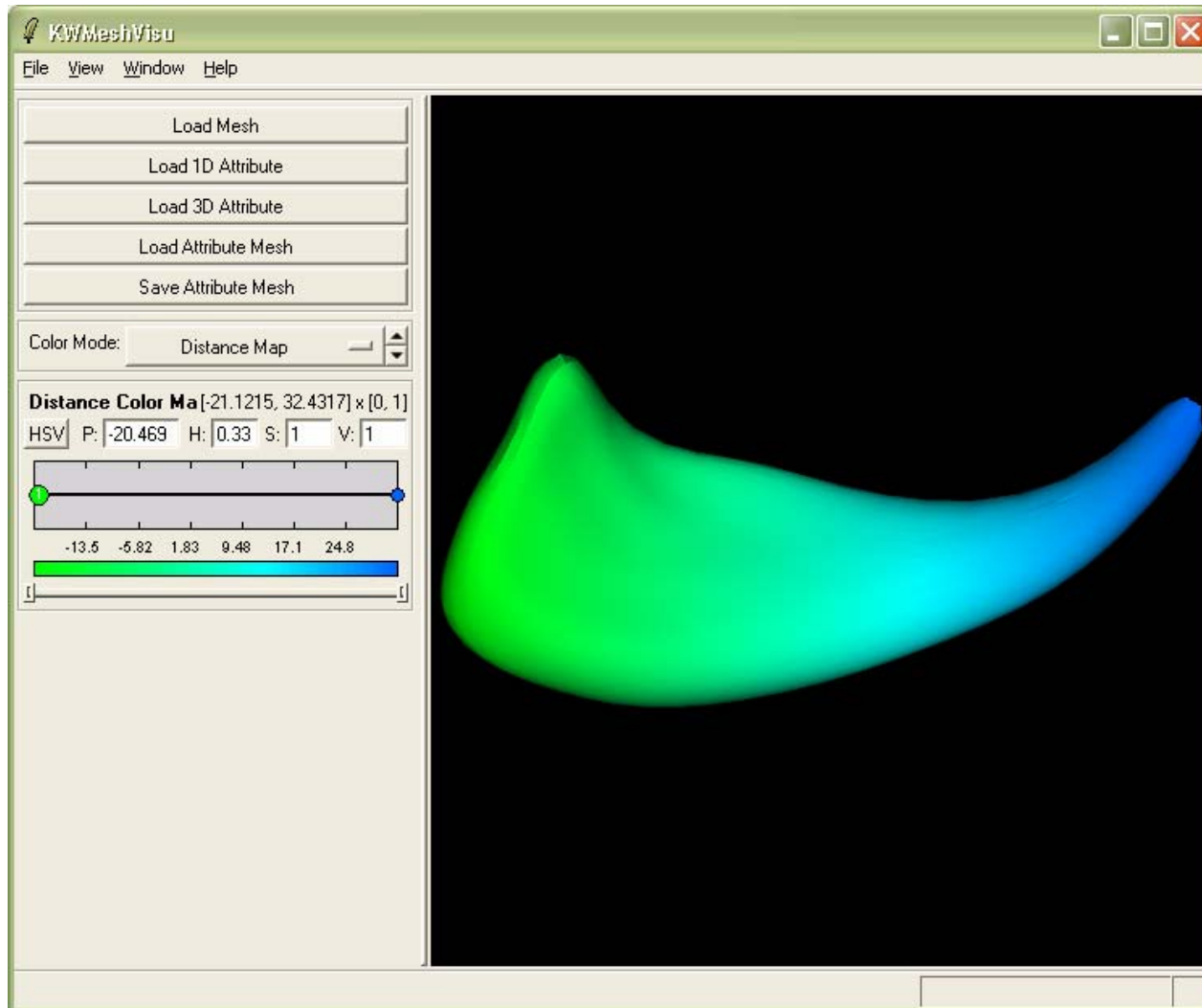
Clinical: Using caudate data for testing.

## Accomplished by end of Programming Week

3D attribute sets – basic functionality and structure for vector fields in place – ten days later, it's completely working

KWWidgets – Most functionality ported to KWWidgets environment; very productive collaboration with Sebastien









# ITK based surface shape analysis modules:

Develop ITK tools for shape representation and statistical shape analysis.

## Team

Ipek Oguz, UNC (Algorithms)  
Christine Xu, UNC (Algorithms)  
Martin Styner, UNC (Algorithms) (contact)

Jim Miller, GE (software)

## Plan/Expected Challenges

Algorithms: Novel scientific visualization for verification and display of the results (distance maps, difference fields, covariance ellipsoid fields, effect size maps, p-value maps etc). Support of SPHARM description in ITK.

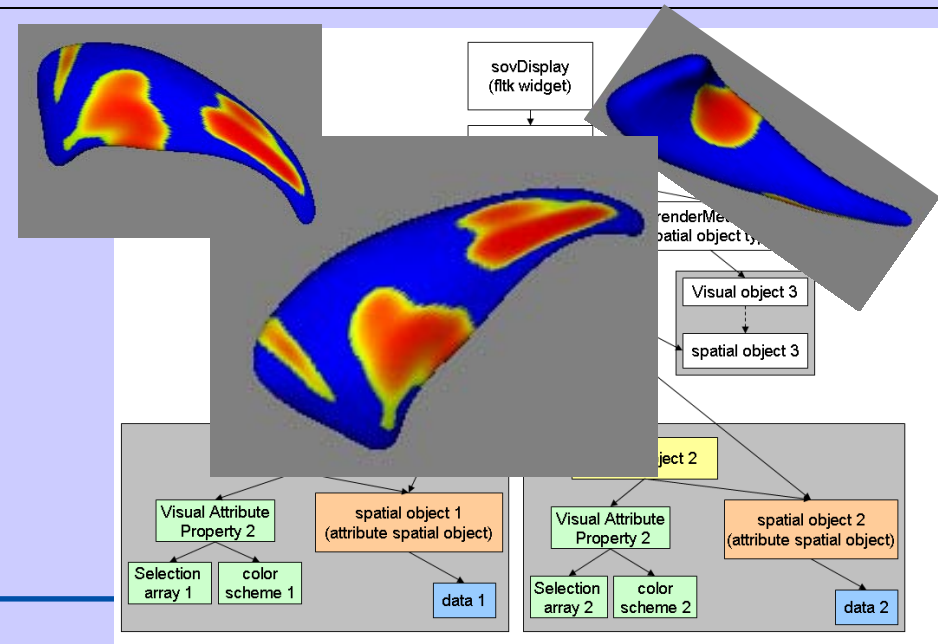
Software: Design support for novel classes in ITK

Clinical: Evaluation in separate Project (UNC-LONI-pipeline)

## Accomplished by end of Programming Week

Design and initial implementation for SPHARM support in ITK, new MeshSource class, Readers/Writers for basic coefficient files, scalar attributed meshes (distance maps, p-value maps). Simple tool for AttributedMesh display.

Source in Sandbox, submission to ITK pending





# Time-Series Navigation for fMRI Statistical Analysis

Develop a Time-Series Navigation algorithm and Client/Server functionalities for fMRI processing

## Team

Sonia Pujol, BWH (algorithms & software)

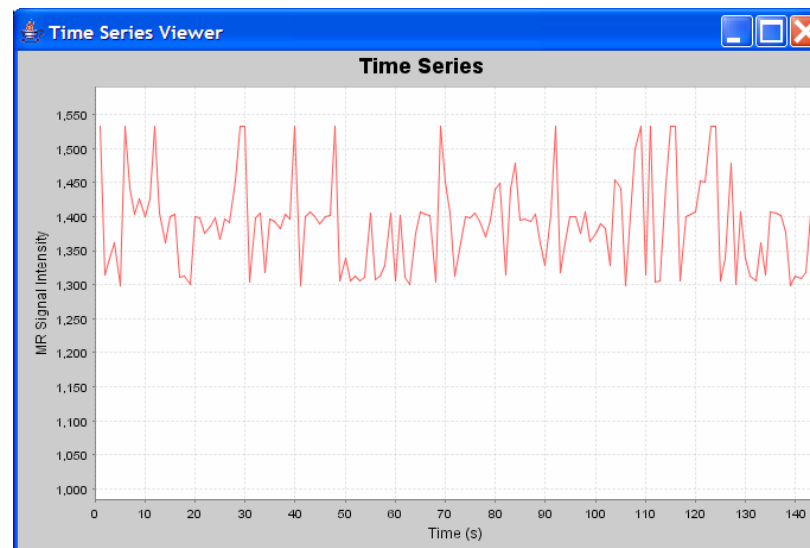
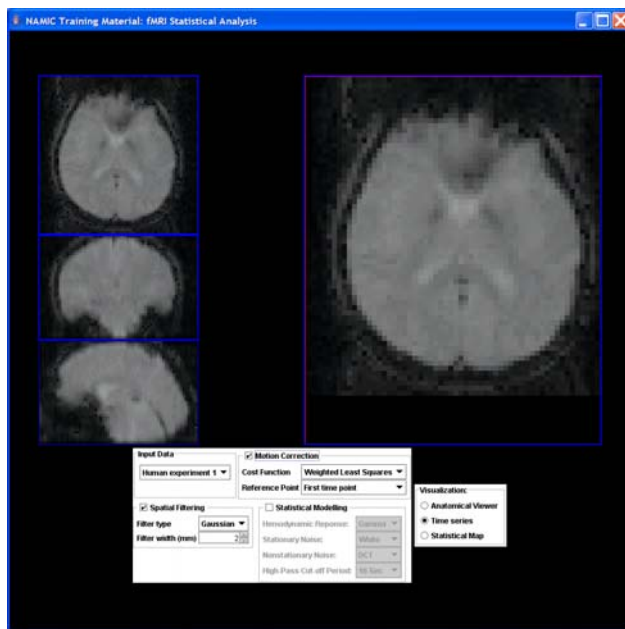
Randy Gollub, MGH (clinical)

## Plan/Expected Challenges/Publication

Algorithms: Develop a client/server 4D application for navigating time-series data

Software: Client-Server Performances. Cross-platform application. Intuitive user interface.

Clinical: Demonstrate the effects of the different pre-processing steps on the results of the statistical analysis of fMRI data.



- Data processing in itk on the Server side.
- 4D Navigation in Time-series data on the Client side
- Cross platform application (java applet)



# Joint Registration and Segmentation:

Develop ITK software module for joint atlas registration and MR segmentation.

## Team

Kilian Pohl, BWH + MIT (algorithms)

Luis Ibanez, Kitware (software)

Karthik Krishnan, Kitware (software)

Jim Miller, GE (software)

Mark Niethammer, BWH (clinical)

## Plan/Expected Challenges/Publication

Algorithm: Start with Powell's method. Try out other optimization methods for increased speed.

Software: Integration of Joint Registration and Segmentation Framework

Clinical: Validate Results with Slicer Version.

Method is going to be published in NeuroImage

## Accomplished by end of Programming Week

- Defined Generic EM Algorithm Structure
  - Generic Statistic classes
  - Image classification classes
- Wrote Registration Framework for Vector Images
  - New image Metric
  - Modify ITK Metric Hierarchy
- Integrated Image Inhomogeneity by Wells 96
- Implemented Segmentation and Registration Algorithm
- Set up Test Scripts

Place your picture/movie here.



# ITK Filter for Affine Invariant Smoothing:

Develop algorithm and ITK filter for affine invariant smoothing.

## Team

John Melonakos, GT (algorithms) (contact)  
Delphine Nain, GT (algorithms)

Jim Miller, GE (software)

Marc Niethammer, BWH (algorithms)

## Plan/Expected Challenges/Publication

Algorithms: start with the smoothing filters currently available in ITK, modify to include an affine invariant version, and refine as necessary.

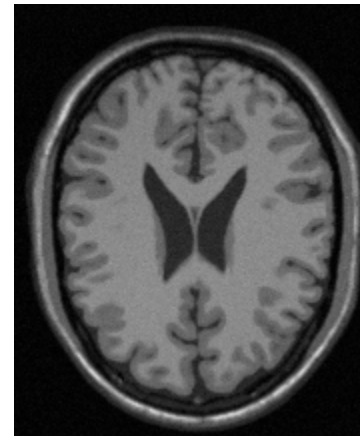
Software: computational speed.

Expected Date for Insight Journal Publication: End of January 2006

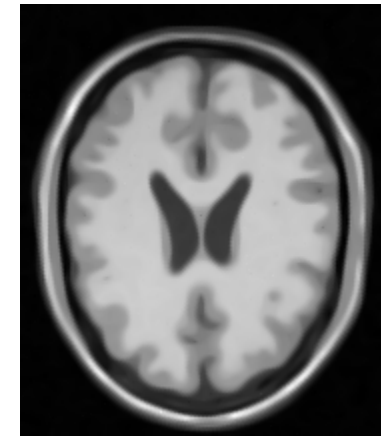
## Accomplished by end of Programming Week

- Implemented [itkAffineInvariantCurvatureFlowImageFilter.h/.ttx](#)
- Implemented [AffineInvariantCurvatureFlowImageFilter.cxx](#) example
- Checked in to the [sandbox](#) under AffineInvariantCurvatureFlow

- Had a great [birthday cake!](#)



Original image



Smoothed image, rescaled  
20 iterations, 0.25 time step



# Riemannian DTI Filters:

Develop algorithms and ITK modules for basic image processing on tensor fields using Riemannian approaches.

## Team

Saurav Basu, Utah (algorithms) (contact)  
Casey Goodlett, UNC (algorithms)  
Tom Fletcher, Utah (algorithms)

Karthik Krishnan, Kitware  
Xiadong Tao, GE

## Plan/Expected Challenges/Publication

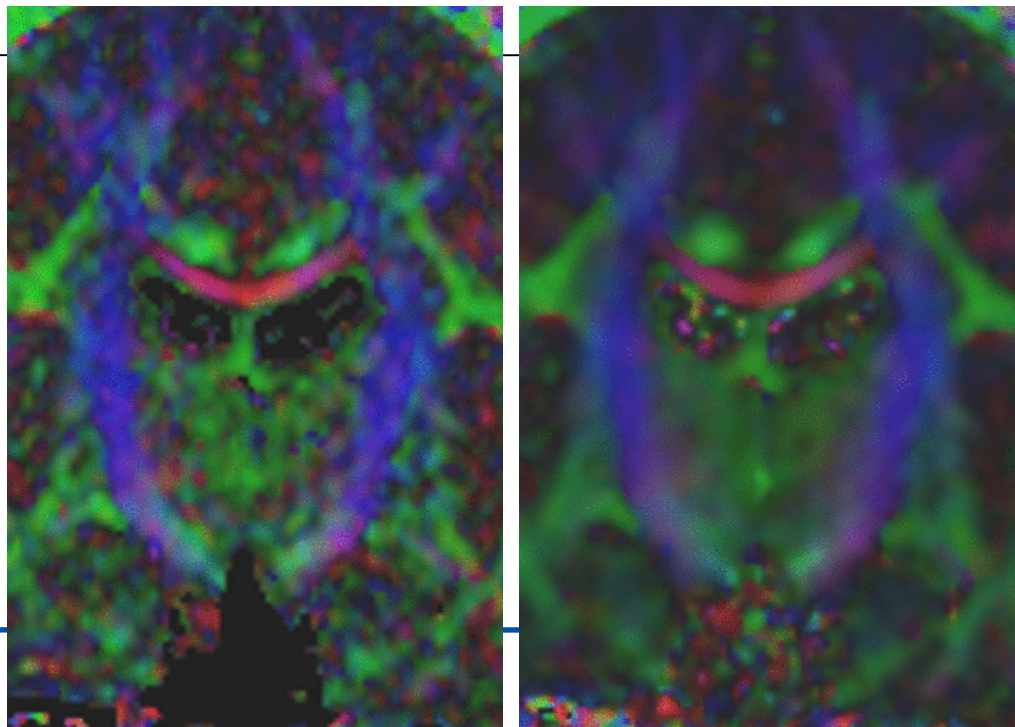
Algorithms: development of Riemannian tensor processing ideas.  
Develop filters for interpolation, resampling, smoothing, etc.

Software: Develop ITK filters that fit well with established frameworks for resampling, interpolation, et c.

Clinical: Validation of results.

## Accomplished by end of Programming Week

- Resolve Tensor/DWI IO Issues
- ITK Registration Framework working on Vector/Tensor Images
- Filters for Euclidean space processing of DTI
- Progress on Symmetric space processing of DTI (Design of classes/filters established)





# Automated Mosaicing and Feature Tracking

Address the limitations of the ITK inverse transform framework.

---

## Team

Tolga Tasdizen and Ross Whitaker (leads)  
Liz Jurrus (feature tracking)  
Paul Koshevoy (automated mosaicking)

## Plan

- Explore ITK and NAMIC for components useful in tracking small, noisy features through 100s of image slices.
- Discuss numeric inverse transform implementation for transforms which do not have an analytic inverse to aid in Mosaicking.

Expected Date for Insight Journal Publication: late 2006

---

## Accomplished by end of Programming Week

Discussed several topics of interest with James Miller, including inverse transform API extension, a bug fix for RegularStepGradientDescentOptimizer and the use of BoxSpatialObject in place an image bit mask as an optimisation to our current mosaicking process.

Also discussed available ITK features to help with feature tracking in biological data, such as the use of the NormalizedCorrelationFilter.



# Angular smoothing and interpolation of DW MRI

Develop an ITK filter to smooth and interpolate DW MR measurements angularly

## Goals

1. Develop an ITK filter to smooth and interpolate DW MR measurements angularly;
2. Test and validate the method by using slicer on the smoothed and interpolated datasets.

## Team

Xiaodong Tao, GE

Dennis Jen, MGH

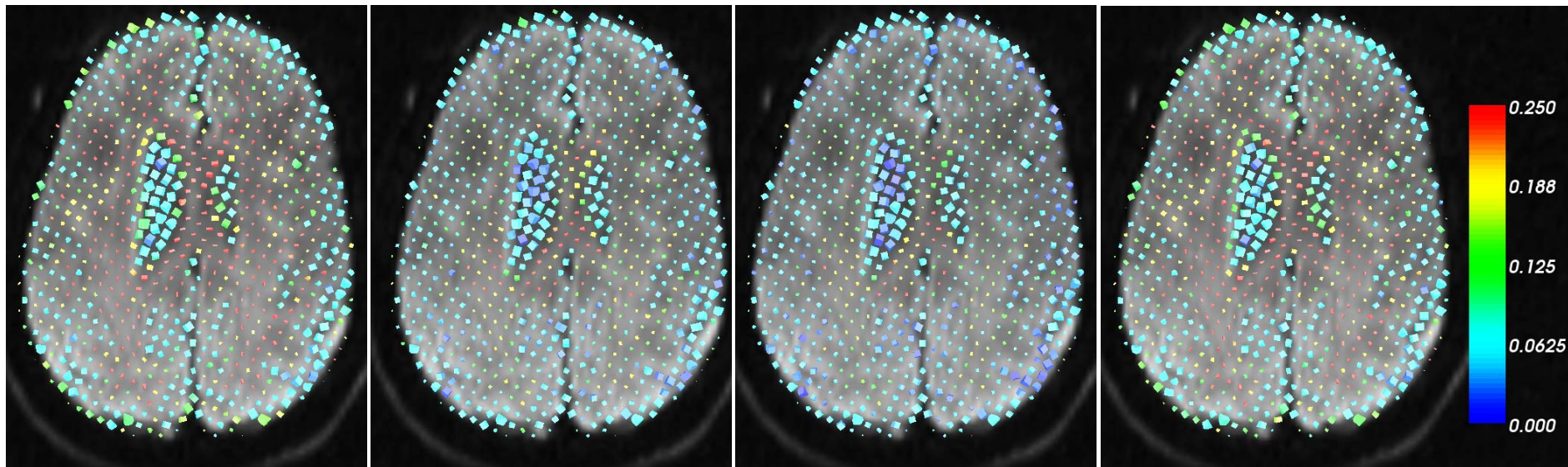
## Plan/Expected Challenges/Publication

**Algorithms:** For each voxel, DW MR measures can be viewed as values defined on the unit sphere. Smoothing and interpolation can then be done in the spherical domain

**Software:** a stand alone program using ITK and an ITK filter

**Validation:** Using Slicer as a platform/tool to compare results based on original DW MRIs and these based on smoothed and/or interpolated DW MRIs.

## Accomplished by end of Programming Week





# MRI Tissue Classification

Develop algorithm and ITK software module for tissue classification of brain-MR images.

## Team

Suyash P. Awate, Utah  
Tolga Tasdizen, Utah (algorithms)  
Ross T. Whitaker, Utah (algorithms)

Dan Blezek (GE)  
Xiaodong Tao (GE)

## Plan/Expected Challenges/Publication

Algorithm: Based on an adaptive, nonparametric model of higher-order Markov statistics.

Publication: [Tasdizen, Awate, Whitaker, Foster: MRI Tissue Classification with Neighborhood Statistics: A Nonparametric, Entropy-Minimizing Approach. MICCAI \(2\) 2005: 517-525](#)

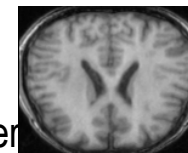
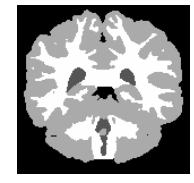
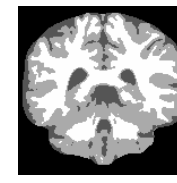
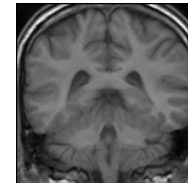
Software: Design and coding of the algorithm virtually complete. **Need help in incorporating some pre-existing automatic ITK-based registration routine as a pre-processing step in our algorithm.** Currently using GUI-based mutual-information-based registration tool in ITK-Applications.

Clinical: Validation – using real IBSR data.

## Accomplished by end of Programming Week

1. Tested Dan Blezek's command-line rigid-registration tool.
2. Added code for doing affine registration. Tested new code. Investigating issues with optimizer and parameter tuning.
3. Decided to redesign code to conform with the planned Markov-modeling framework in ITK. Need to cooperate with Kilian, Jim Miller

IBSR data: Validation of classification method



IBSR data

Classification

IBSR Ground Truth





## Bone Meshing:

Develop meshing algorithms suitable for surface contact analysis. Integrate these algorithms and required user interaction into Slicer.

### Team

Vincent Magnotta, Iowa (contact)

Kiran Shivanna, Iowa

Nicole Grosland, Iowa

Steve Pieper, Isomics (software)

Alexander Yarmarkovich, Isomics (software)

### Plan/Expected Challenges/Publication

Algorithms: implement basic voxel mesh structure including modified node numbering and material property definitions.

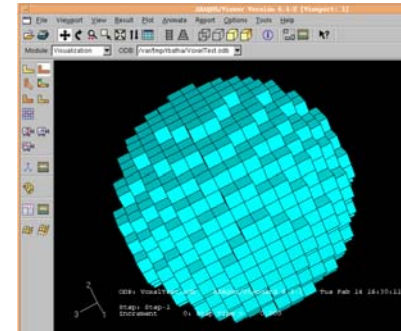
Software: Add visualization of resulting meshes into Slicer. Work on adding Slicer user interaction features that will be utilized by additional meshing algorithms in the future.

Expected Date for Insight Journal Publication: April 1, 2006

### Accomplished by end of Programming Week

We have created a new ITK filter called ImageToHexahedronMeshFilter which derives from ImageToMeshFilter. The filter allows the user to use the voxel coordinates for node numbering and specify a per voxel material properly assigned using CellTraits. Currently the material property generates an elastic modulus assuming the data is a CT image. This will be generalized to a user defined function. A writer to write out the Mesh into an Abaqus input deck has been written.

### Output of Filter Shown in Abaqus





# GPU Tube Visualization for the Slicer Tractography Module

## Team

Steve Pieper, Isomics (algorithms) (contact)

Thomas Lechner, Northwestern (software)

Nicole Aucoin, BWH (compiling with VTK 5.1)

Lauren O'Donnell, MIT (debugging DTMRI)

Mathieu Malaterre, KW (VTK guru)

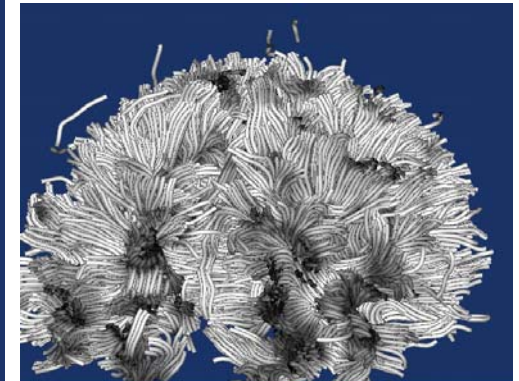
## Plan/Expected Challenges/Publication

Algorithms: a simple filter which converts all lines in the input PolyData to degenerate quads. A vertex program puts the vertices of the quad at the right place, and then a fragment program renders this quad like a curved tube surface.

Software: integrate tube visualization with tractography module in slicer; this visualization currently requires vtk 5.0

## Accomplished by end of Programming Week

- Compiled Slicer with Vtk 5.1
- Some debugging of DTMRI (needs more)
- Some (untested) integration code





# Block Matching Registration:

Coding into itk Block Matching Registration Algorithm published in MICCAI'02, nowadays in matlab.

## Team

Eduardo Suárez, ITC (algorithms) (contact)  
Izzat Sabaggah, ITC (algorithms)

## Plan/Expected Challenges/Publication

Algorithms: code into itk Block Matching Registration Algorithm published in MICCAI'02, nowadays in matlab.

Software:

Clinical:.

This is a starting work thought to be added to itk

Expected Date for Insight Journal Publication:

## Accomplished by end of Programming Week

Skeleton of the block matching filter already defined for one pyramidal level, w/o weighting.

## (Add picture to illustrate accomplishment)

Place your picture/movie here.



# Slicer 3 Architecture:

Allow components of Slicer 3 project to work well together.

## Team

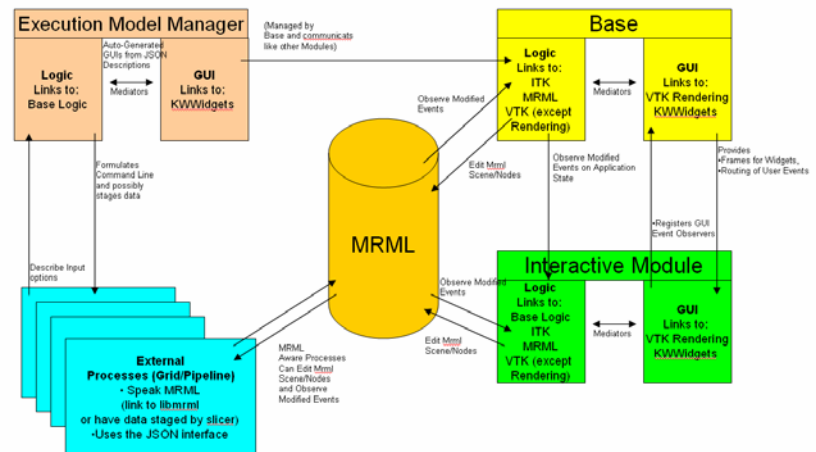
Steve Pieper  
 Bill Lorensen  
 Ron Kikinis  
 Mike Halle  
 Noby Hata  
 With Slicer 3 Team

## Plan/Expected Challenges/Publication

- Large Complex Application
- Distributed Project Team
- Most Developers Part-Time
- Leverage Existing Code Base
- Deliver Platform Quickly
- NA-MIC Friendly License
- Maximum Code Reusability

## Accomplished by end of Programming Week

- MRML Functionality and VTK/ITK Syntax Prototypes
- Execution Model Consensus
- GUI Toolkit Prototyping and License Due Diligence





# Slicer3 Architecture Discussion:

develop a user and developer friendly application environment

## Team

All (algorithms)

All (software)

Presenters: Ron Kikinis, Steve Pieper, Mike Halle,  
Tina Kapur, Will Schroeder, Berk Geveci, Mike Pan,  
Bill Lorensen, Luis Ibanez

All (clinical)

## Plan/Expected Challenges

Algorithms: express needs for rapid development,  
experimentation, and interactive steering/visualization

Software: flexible software framework, easy to use

Clinical: friendly user interface, full featured

**Discussions:** Big Picture, Technical,  
Developer Empowerment, Reuse, KWWidgets,  
3D Widgets, Server Manager, Pipeline, Grid,  
Licenses...



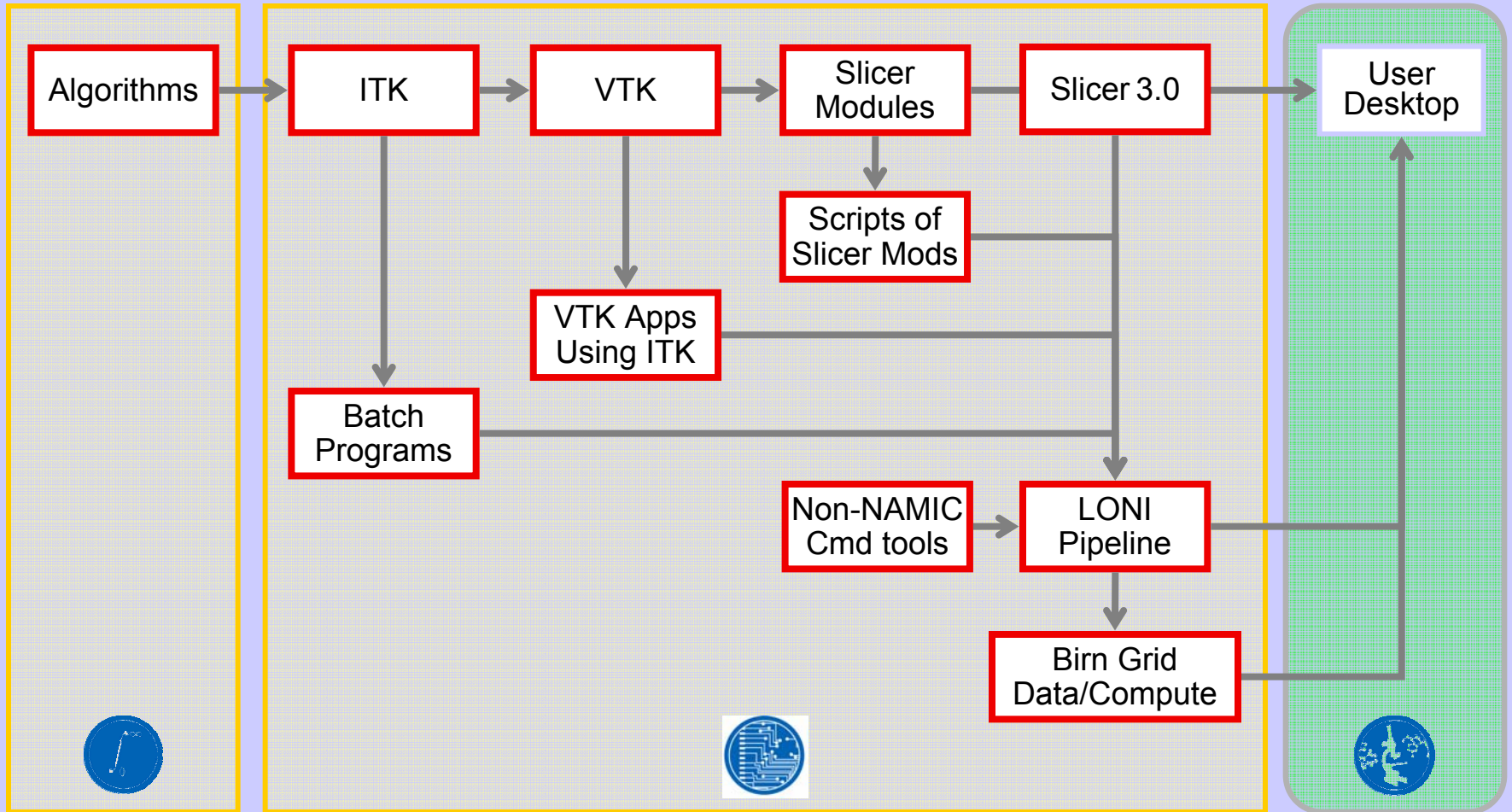
## **Plans:**

- \* Algorithm Wrapping ( ).
- \* Widget "Bake Off"
- \* Initial Focus on Image Viewer / Segmentation Editor





# Slicer 3.0





# Slicer 3: Web Presence:

Update infrastructure to support Slicer 3's web presence.

## Team

Nicole Aucoin, BWH (contact)

Andy Cedilnik, Kitware (software)

No core 3 participant needed.

## Plan/Expected Challenges/Publication

Algorithms: n/a

Software: download stats script, email, cvs access form, license agreement form (with automated d/l selection?), wiki?

Clinical: n/a

## Accomplished by end of Programming Week

Andy and I supported other projects during the Programming Week, but have continued to work on our to-do list, finishing the cvs write access request form by end of Jan. The bug tracker will be updated over the first week of Feb.

Request for Slicer Write Access – Mozilla

File Edit View Go Bookmarks Tools Window Help

Back Forward Reload Stop <https://www.na-mic.org/Admin/SendPassword.cgi> Print

Home Bookmarks post to delicio.us my delicio.us

Dashboard... Slicer.Wo... CVS--C... slicer2/M... phpBugTr... 3D Slicer Request f... x

### Request for Slicer Write Access

This form is to be filled out to request write access to the [Slicer CVS repository](#). It is only to be used by developers who wish to contribute code to the repository, as anonymous read only access is allowed via commands outlined at [slicer.org](#). The password is encrypted and sent to the system administrators, and is not recoverable.

#### Send password

Please fill out this form to send an encrypted password to Slicer system administrators. Under the "Reason" field, please enter information about what the request is for. If there is no clear indication what the request is for, it will be ignored.

Full Name

Institution

Desired UserName

Phone Number

E-Mail

E-Mail repeat

Password

Password repeat

Reason:

Submitting code to the Slicer CVS repository constitutes agreement to the [Slicer license](#).  
I have read, understood, and I agree to the license terms:

Send



# Execution Model:

Develop API and standards enabling command line modules to be executed in Slicer/Grid/Cluster contexts.

## Team

Dan Blezek, GE (Engineering) (contact)  
Brendan Faherty, UCSD (Engineering)  
Jeffrey Grethe, UCSD (Engineering)

## Plan/Expected Challenges/Publication

- evaluate needs across command line API, grid, cluster contexts
- find/develop/augment API for command line apps
- define standard for command line / GUI communication
- prototype using simple registration executable

## Accomplished by end of Programming Week

- Developed straw man of description format
- Adopted JSON as *de facto* standard
- Garnered buy in from Slicer, LONI Pipeline, Grid

## Execution Model Working Group







# Slicer 3 Data Model:

Design a Modular service for providing a centralized access to data required by Slicer.

## Team

Mike Halle, BWH (algorithms) (contact)  
Alex Yarmarkovich, BWH (algorithms)

Xiandong Tao, GE (software)  
Luis Ibanez, Kitware (software)

## Plan/Expected Challenges/Publication

Algorithms: Identify the Use Cases of a Data Model, and the services that it should provide to applications.

Software: Safe implementation (some applications will be IGS) robustness, speed, memory access considerations. A prototype will be developed during the programmers week in order to provide a proof of concept.

This effort is part of the re-architecture plan for Slicer 3.0.

Expected Date for Insight Journal Publication: January 31<sup>st</sup> 2006

## Accomplished by end of Programming Week

- Described feature sets. [http://wiki.na-mic.org/Wiki/index.php/Slicer3:Data\\_Model](http://wiki.na-mic.org/Wiki/index.php/Slicer3:Data_Model)
- Formulated requirements to MRML API, created use cases, identified issues [http://wiki.na-mic.org/Wiki/index.php/AHM\\_2006:ProjectsSlicerDataModel#Slides](http://wiki.na-mic.org/Wiki/index.php/AHM_2006:ProjectsSlicerDataModel#Slides)
- Created prototype for a Path based scene description.
- [http://wiki.na-mic.org/Wiki/index.php/Slicer3:MRML3\\_Path](http://wiki.na-mic.org/Wiki/index.php/Slicer3:MRML3_Path)
- Created Slicer3 module inside Slicer3 for MRML API prototype



# Slicer 3 Interface Design:

Design a new interface for Slicer3 that strongly supports both user needs and technical requirements.

---

## Team

Wendy Plesniak, BWH (contact)  
Michael Halle, BWH  
SebastienBarre, Kitware  
Mathieu Malaterre, Kitware

## Plan/Expected Challenges/Publication

Software:

- Promote user-centered design and usability engineering in the Slicer3 development effort;
- Determine user and technical needs from core user communities;
- Create a consistent and centrally-prescribed look & feel;
- Design Slicer's GUI architecture and its interface to application layer;
- Specify and develop set of custom widgets for Slicer3;
- Implement GUI layer;
- Conduct user testing and revise designs;
- Document coding paradigms for developers.

---

## Accomplished by end of Programming Week

- Made final decision to develop Slicer3 GUI based on KWWidgets;
- Adapting LONI and Kitware approaches to prescribing application-wide look & feel for Slicer3;
- Experimenting with KWWidgets;
- Conducting early discussions with users about technical requirements;
- Designed draft of Slicer GUI architecture;
- Designed draft mechanism for interfacing GUI layer to logic layer (will require testing).
- Currently implementing this draft design (C++);



# Slicer 3 – IGSTK Integration:

Use the IGSTK library from Slicer in order to provide the basic support for IGS applications.

## Team

Nobuhiko Hata, BWH (algorithms) (contact)  
Steve Pieper, Isomics (algorithms)

Patrick Cheng, ISIS Center, Georgetown (software)  
Luis Ibanez, Kitware (software)

## Plan/Expected Challenges/Publication

**Algorithms:** Identify the modifications that would be needed in Slicer and IGSTK in order to use the IGSTK library from Slicer.

**Software:** remove from IGSTK software dependencies that may conflict with the software framework of Slicer.

Attempt linking Slicer with IGSTK and using some of its classes  
Explore the option of using a State Machine for managing the workflow of a Slicer based IGS application.

This effort is part of the re-architecture plan for Slicer 3.0.

Expected Date for Insight Journal Publication: January 31<sup>st</sup> 2006

**Accomplished by end of Programming Week**

**(Add picture to illustrate accomplishment)**

Place your text here.

Place your picture/movie here.



# Large Scale Job Submission and Slicer 3

## Large Scale Job Submission Integration with Slicer3 Execution Model

### Team

Brendan Faherty, UCSD (Grid Infrastructure) (contact)  
 Jeff Grethe, UCSD (Grid Infrastructure)

Dan Blezek, GE (Software)  
 Steve Pieper, BWH (Software)

NAMIC Community (application use cases)

### Plan/Expected Challenges/Publication

Goal: Dynamically enable staging and execution of NAMIC algorithms through Slicer 3 execution model and UCSD grid interface to execution model.

Grid Infrastructure: Develop API for Grid Interface to enable Slicer3 executable command line communication with Condor. Expand upon 'namic-submit' strawman functionality.

Software: Slicer3 execution model. (related programming week project)

Clinical: Collect two distinct use cases which provide meaningful data for the NAMIC community.

### Accomplished by end of Programming Week

Established preliminary Slicer / Executable JSON language specification. Decided upon a single, defined execution model for all external Slicer communication to executables, pipelines and grids.

Added JSON language library to gi (Grid Interface) program and built out Condor data structure translation. Streamlined Slicer to Grid program logic to eliminate intermediate steps (one data transmission will perform Grid Execution).





# Large Scale Algorithm Job Submission via Condor: *Enable distributed execution of NAMIC algorithms.*

---

## Team

Jason Gerk, UCSD (Grid Infrastructure) (contact)  
Jeffrey Grethe, UCSD (Grid Infrastructure)

Steve Pieper, BWH (Grid Applications)

NAMIC Community

## Plan/Expected Challenges

**Grid Infrastructure:** Deploy the infrastructure and software to enable the scheduling and submission of staged NAMIC algorithms.

**Scalability:** Under heavy usage, monitor connections and performance. The discovered bottlenecks will aid in configuration improvements for specific implementations.

**Grid Applications:** Application of the infrastructure to specific NAMIC algorithms.

**NAMIC Community:** Collect use cases and requirements as to how this infrastructure can be utilized by NAMIC developers and researchers.

## Accomplished by end of Programming Week

Gained understanding of how NAMIC would like to submit jobs. Different that how Condor works out of the box, so we'll have to customize for NAMIC community.

Ran actual tests using NAMIC datasets and Registration executable.

Uncovered a few global configuration issues for Compute Cluster (non-UID, memory, versions).

Built preliminary shell scripts that when passed to cluster node will download datasets and executable, run program, and push results back into BIRN.

Built sample shell script that can be run command line which handles automatic job submission by creating submit files (and multiple submissions) and passing arguments into executable.

Worked with Steve to draft 'Straw Man' for proposed design of 'namic-submit' functionality and syntax. Should handle proxy, submission, required files, logs, stdout, stdin, and job tracking.





# Itku: Command-line interface to ITK

Use Teem's command-line parsing library to build a command-line interface to ITK .

## Team

Raul San Jose (BWH)  
Gordon Kindlmann (BWH)

Itk Experts help/feedback

## Plan/Expected Challenges/Publication

Algorithms: n/a

Software: Implementation of the communication between Teem and ITK through a nrrd-itkImage interface: **rapid prototyping**  
Clinical: n/a

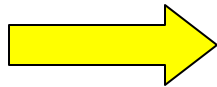
This is a continuation from the last Programming Week.

Expected Date for Insight Journal Publication:

## Accomplished by end of Programming Week

### Utilities:

itkio.cxx  
itkuMacro.h  
Nrrditk.txx  
Itknrrd.txx



### Plugin:

Myfilter.cxx: →

MyfilterHandler.txx: →

### itkuLoad:

- parse command line: **hest**
- Find pixel type and dim: **itkio**
- Template expansion: **itkuMacro**

### Tempate<ImageType>

- Wrap nrrd into itk: **nrrditk**
- Execute itk pipeline
- Wrap itk into nrrd: **itknrrd**

**itku myfilter -i a.nrrd -o b.nrrd**



# ITKu:

Minimalist command line tools using ITK .

## Team

Raul San Jose, BWH (collaborator) (contact)  
Gordon Kindlmann, BWH (collaborator)

Jim Miller (GE)

## Plan/Expected Challenges

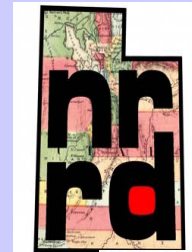
Algorithms: N/A.

Software: nD array manipulation and  
pipeline communication between ITK filters.

Clinical: Command line versatility to accomplish tasks.

## Accomplished by end of Programming Week

- Team builds with Cmake
- Define strategy for itku
- Filter example: dynamic loading of  
itkDiscreteGaussianImageFilter





# NrrdIO + measurement frames:

Update NrrdIO in ITK and possibly add a measurement frame to ITK

## Team

Gordon Kindlmann, BWH  
Raul San Jose Estepar, BWH

Bill Lorensen, GE ?

## Plan/Expected Challenges

1. Cvs commit latest NrrdIO into ITK (so that new “measurement frame” field for DWI can be parsed)
  2. Make Slicer parse updated DWI NHDR headers
  3. Help others w/ DWI data write NHDR headers
  4. Mapping from the NRRD fields to ITK image fields
  5. Experiment with Measurement Frame in ITK Image
- ⇒ Have to learn more about ITK

Software: ITK Image data structures

Clinical: save the children

## Accomplished by end of Programming Week

1. Latest NrrdIO now committed into ITK; orientation info properly handled
2. Slicer can parse updated DWI NHDR headers (but via a VTK, not ITK, reader)
3. Rough spec of needs for DWI ITK Image type; working with Karthik Krishnan and others to get this implemented

### TODO:

- Mapping NRRD<-->ITK non-scalar images
- Experiment with Measurement Frame in ITK







# LONI Pipeline

## Team

Michael Pan  
Jagadeeswaran Rajendiran

## Plan/Expected Challenges/Publication

Work through various issues with actual use cases

- Martin Styner
- Lauren O'Donnell
- other interested parties

## Accomplished by end of Programming Week

Worked with/introduced various people to Pipeline

- Martin Styner (see [UNC shape analysis slides](#))
- Katharina Quintus
- Lauren O'Donnell
- Slicer (see [Slicer slides](#))
- Wendy

Worked through various bugs and issues

- Program side effects (included in beta6.2)
- Argument value virtualization
- Default argument values
- Ability to curtail number of spawned processes on client side (included in beta 6.2)

Open issues

- GUI bugs (In progress)
- program responsiveness
- debugging issues
- Matlab support
- Naïve user support
- Pipeline JSON argument value binding (First draft in progress)
- Simplified STUDIO access
- Windows Cygwin support



# LONI pipeline for UNC shape analysis:

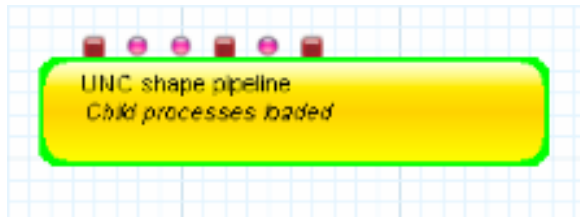
Developing LONI pipelines for UNC shape analysis ready for clinicians

## Team

Martin Styner, UNC

Michael Pan, LONI

Jagadeeswaran Rajendiran, LONI



## Plan/Expected Challenges/Publication

Algorithms: create LONI pipeline modules and network for new set of shape analysis executables, as well as interface for running UNC shape analysis in LONI as simple as possible. Continuation of last programming week.

Software: Training, Extension of LONI software if needed.

Clinical: Validation of results.

## Accomplished by end of Programming Week

LONI pipeline for full processing created

Bug fixing and considerable feature request issued

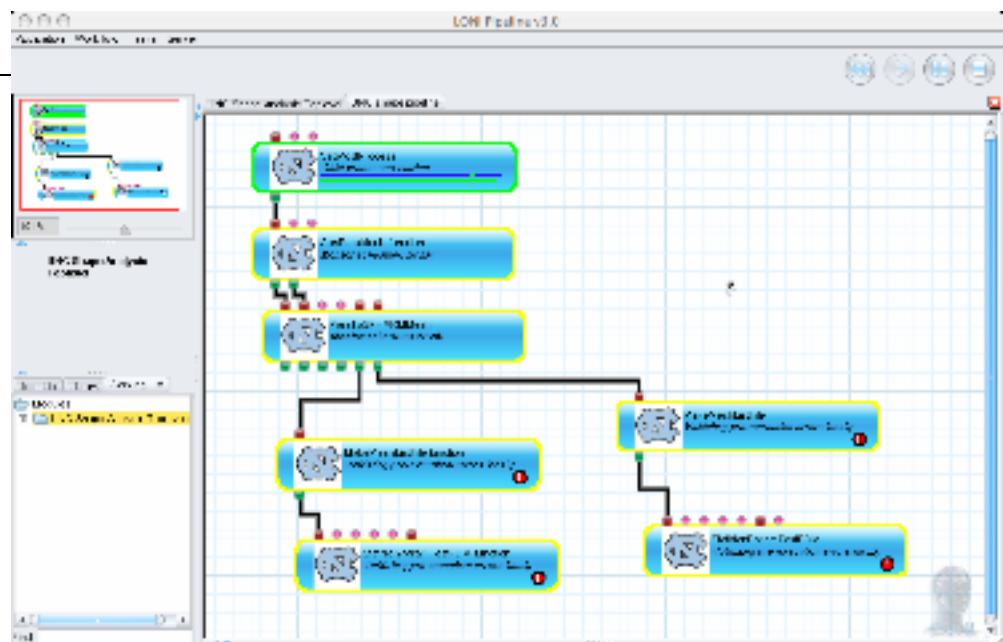
Top-level view encapsulating details from users

Nearly ready for Core 3 and training core

Open Issues: Full encapsulation, job distribution,

Visualization of Quality Control data and results,

Validation of Results





# LONI Pipeline for UNC shape analysis:

Move the UNC script based shape analysis pipeline to the LONI pipeline for enhanced applicability.

## Team

Martin Styner, UNC (Algorithms) (contact)

Ipek Oguz, UNC (Algorithms)

Jags Rajendiran, UCLA (software)

## Plan/Expected Challenges

**Algorithms:** Use the existing algorithms and fit them into the LONI pipeline architecture.

**Software:** If needed, any adaptations are made.

**Clinical:** Validation of results from both UNC based pipeline to LONI pipeline applied to NAMIC Harvard VA caudate dataset.

## Accomplished by end of Programming Week

- Pipeline constructed and validated on individual datasets
- Feature necessity and bug report list for LONI
- Combination of LONI and UNC Software
- Results from pipeline validated

