

---

# FSurf: A FreeSurfer Analysis Service for OSG

OSG All Hands Meeting  
March 7, 2017

---

**Suchandra Thapa**  
Robert Gardner  
University of Chicago

Donald Krieger  
University of Pittsburgh

# FreeSurfer

---

Widely used software suite for analysis of human brain MRI scans. It has been used for studying human brain anatomy in a variety of contexts such as studying the neurophysiology of depression, examining possible anatomical differences involved in ADHD, and studying autism.



# FreeSurfer continued

---

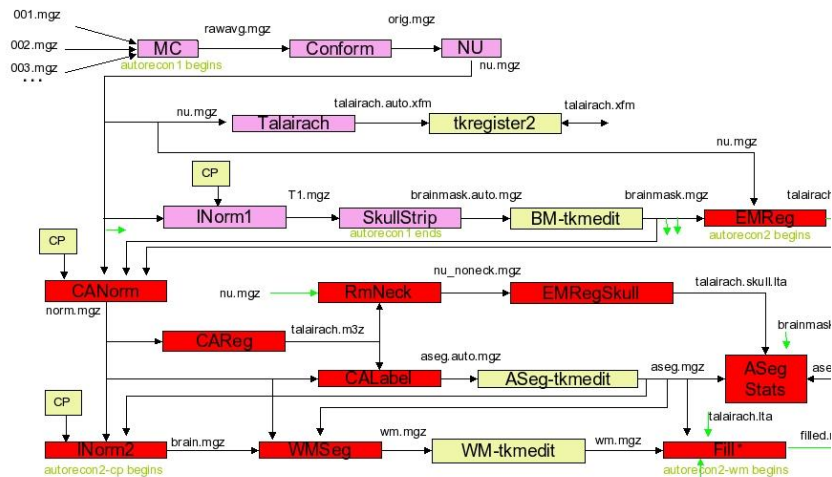
- Users submit MRI scans and after processing receive annotated and labeled images
- Most users run a standard workflow provided by FreeSurfer
- Some users have more advanced usage:
  - Multiple inputs to a single workflow
  - Different parameters for various workflow stages

# Standard FreeSurfer workflow

---

- Can be split into 4 main stages:
  - autorecon1 - MRI image preparation and normalization
  - autorecon2-volonly - further normalization and image preparation
  - autorecon2 - same as prior step as well as generating initial surfaces for brain regions
    - Can be run on left and right hemispheres in parallel
    - Makes effective use of multiple cores (using OpenMP)
  - autorecon3 - final identification and labeling of brain regions
- Running all three stages takes 16-24 hours on a single core, 4-8 hours with 8 cores

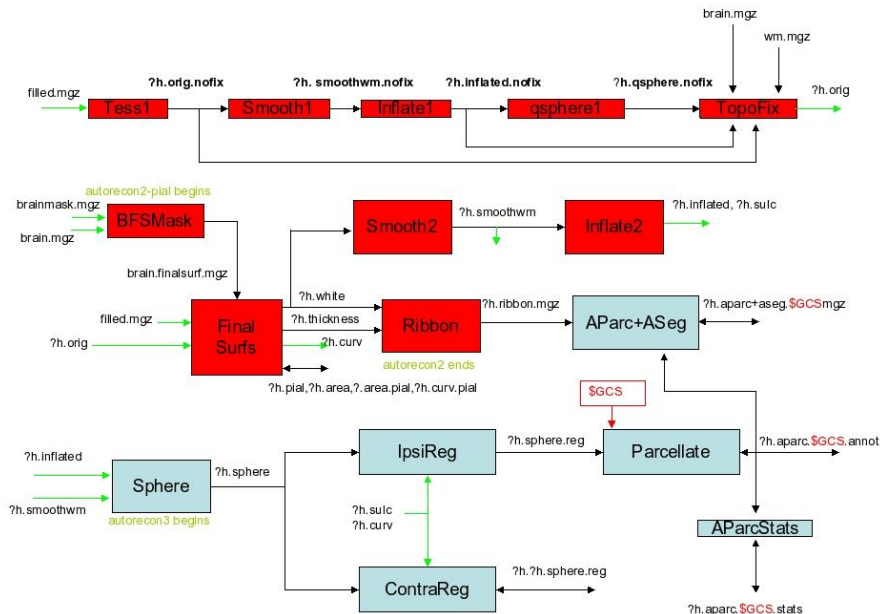
# FreeSurfer stages



WMSeg includes mri\_segment, edit\_wm\_with\_aseg, and mri\_pretext

CP = Control Points

Fill can have (aseg.mgz&tal.1ta) or (tal.xfm, cutting planes) as input, but not both



# Goals for the FSurf service

---

- Provide a service that allows FreeSurfer users to take advantage of OSG resources to run FreeSurfer workflows
- Allow users to be able to access the service without having to obtain OSG accounts or requiring knowledge of DHTC concepts
  - Analogous to the Galaxy service for BLAST users
- Allow users quickly start using the service (ideally within an a few hours of signing up)

# Processing requirements

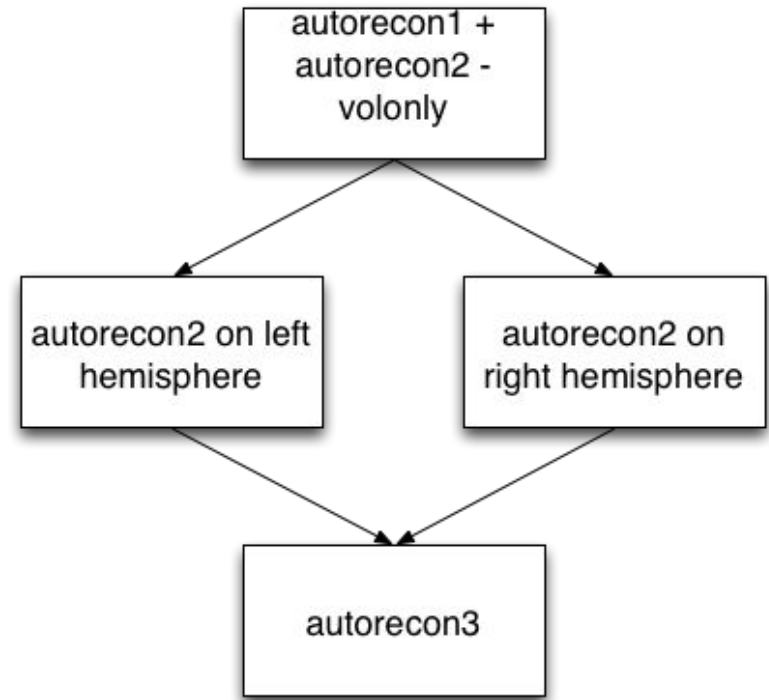
---

- I/O requirements:
  - input consists of file with MRI data in *compressed* format (~10-40MB)
  - output consists of multiple files in a subject directory (~200-300MB tar.bz2 file)
- Memory
  - Dependent on MRI input about ~1.8 to 3GB of RAM
  - May be more for outliers
- Processing
  - ~6-8 hours on a dedicated node with 8 cores
  - portions of the workflow can use multiple cores effectively

# Basic FSurf workflow

---

- Runs the 4 stages in a way that takes advantage of parallelism
  - autorecon1 + autorecon2 -volonly
    - In parallel:
      - autorecon2 on left brain hemisphere (using 8 cores)
      - autorecon2 on right brain hemisphere (using 8 cores)
    - autorecon3
  - Only the two parallel steps can effectively use multiple cores
  - Can be run with multiple scans of same individual for better error correction





# More advanced workflows

---

- Multiple input workflows:
  - FSurf can run the basic workflow using multiple scans of the same individual
  - Allows for errors or motion in scans to be corrected automatically
- FSurf can run arbitrary FreeSurfer workflows
  - Users indicate which options to give to FreeSurfer when running workflow
  - FSurf will then run FreeSurfer with just those options instead of using the basic workflow

# End user interface

---

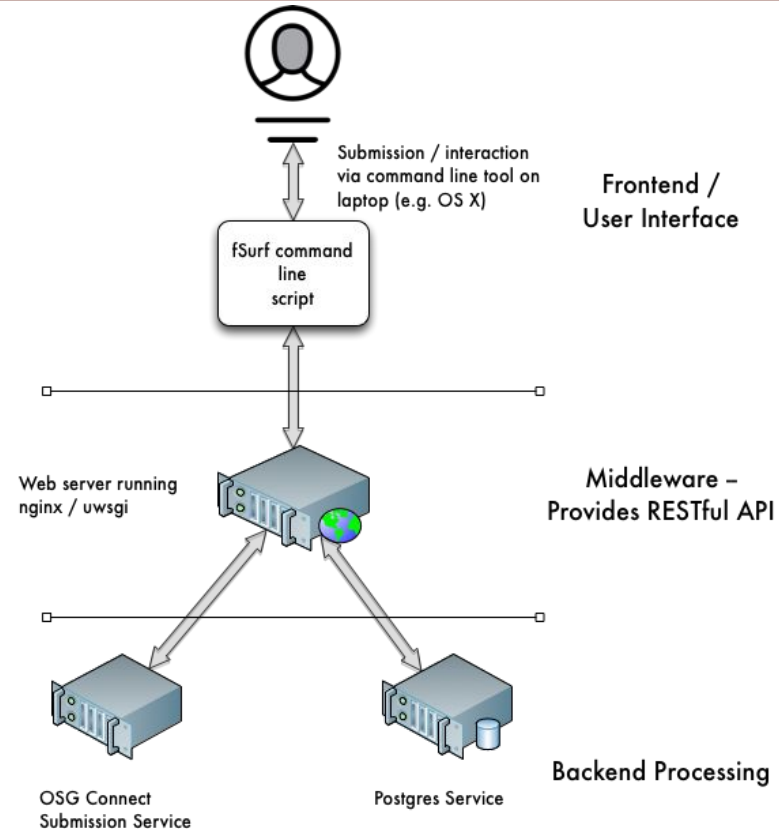
- Simple, easy to use script
  - No installation needed, just download the FSurf script
- Can be used on linux/unix or OS X systems
- Example commands

```
> fsurf submit --input=SubNo_01_defaced.mgz  
> fsurf status --id 4302  
> fsurf output --id 4302
```

- User notified by email when submitted workflows are completed

# FSurf technical architecture

- Front-end user interface
  - Command line script
    - Communicates to the execution service via RESTful-API commands
    - Allows user to submit, view, and remove workflows as well as download results
  - Middleware
    - Nginx / uwsgi server provides RESTful API
- Backend services (running on a single VM)
  - Postgres database
    - Tracks workflow status, locations of inputs and results
    - Stores user information
  - Pegasus submission service (using same infrastructure as OSG Connect) to

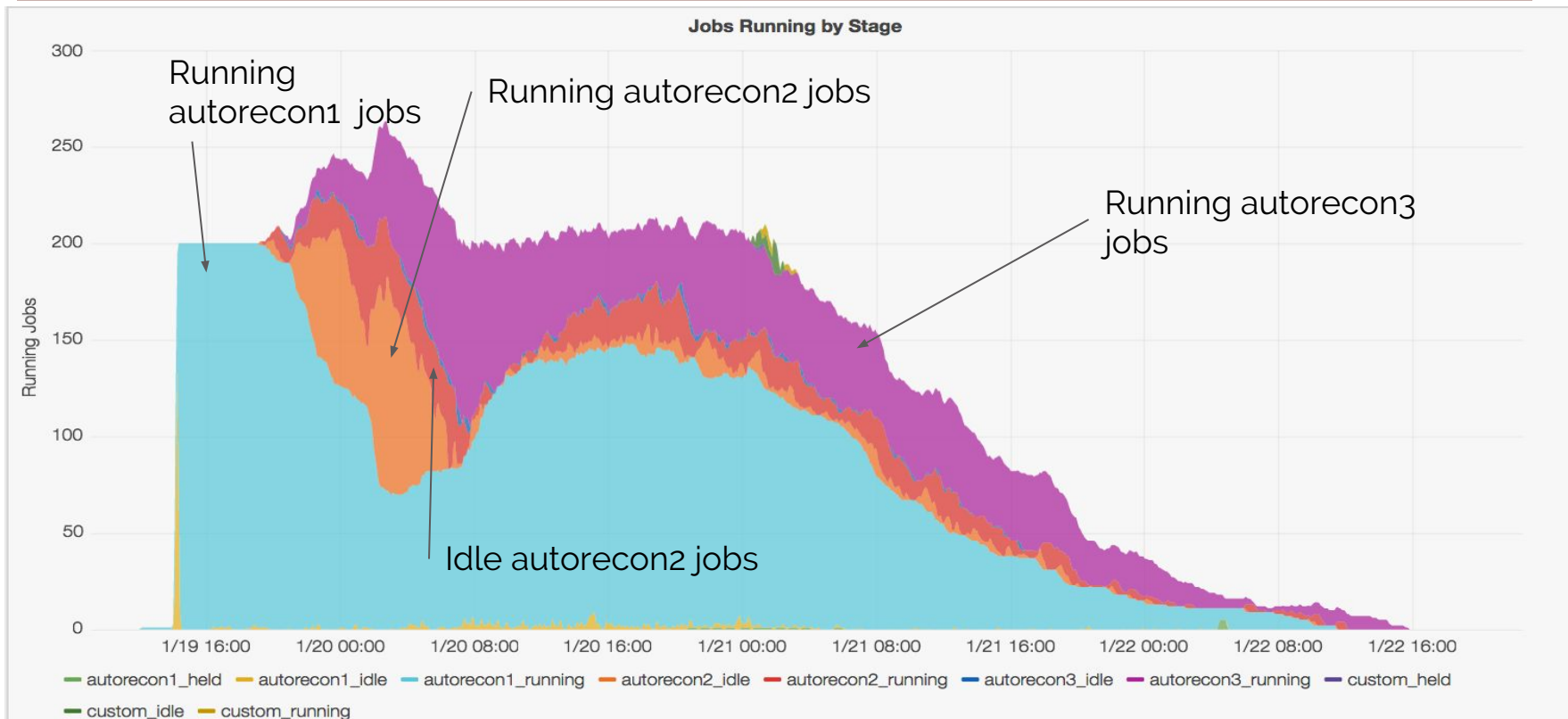


# Monitoring workflow status

---

- Server handling FSurf job submission and workflow management instrumented with probes to monitor workflow status
- Information sent to a graphite server and available as a Grafana dashboard

# Monitoring workflow status 2



# Testing and Validation

---

- FSurf has been fairly extensively tested and validated
- Spent several months submitting large numbers of workflows using various options and validating that the outputs match reference outputs
- Discovered several limitations in infrastructure and software and corrected them during testing
- Current service can reliably handle and process large submissions

# Jenkins for QC

## Automated testing and validation of FSurf services

- Up
- Status
- Changes
- Workspace
- Build Now
- Delete Project
- Configure
- Move
- GitHub
- Embeddable Build Status

### Build History

find

#78	Mar 3, 2017 8:53 PM	74 KB
#77	Mar 3, 2017 8:53 PM	66 KB
#76	Mar 3, 2017 8:53 PM	68 KB
#75	Feb 25, 2017 3:00 AM	19 KB
#74	Feb 25, 2017 3:00 AM	19 KB

[RSS for all](#) [RSS for failures](#)

### Project MRN\_1 custom check

Full project name: Freesurfer tests/MRN\_1 custom check  
Analyze MRN\_1 file using freesurfer and compare to reference output

[edit description](#)

[Disable Project](#)



Project disk usage information + trend graph

**Disk Usage:** Workspace 840 MB (On slaves 840 MB, Non slave workspaces -), Builds 245 KB (Locked -), Job directory 287 KB

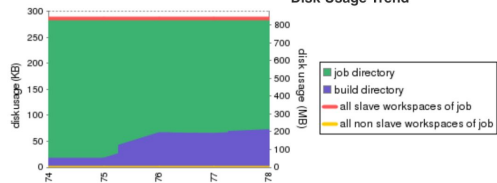
### Upstream Projects

[Freesurfer Testing - FSurf Beta Test](#)

### Permalinks

- Last build (#78), 2 days 13 hr ago
- Last stable build (#78), 2 days 13 hr ago
- Last successful build (#78), 2 days 13 hr ago
- Last completed build (#78), 2 days 13 hr ago

### Disk Usage Trend



# Release status

---

- Released initial version of the service with support for standard workflow
- Released second version of service with support for running standard workflow with added features and for running advanced workflows
- **Public release imminent!**



# Conclusions

---

- FSurf offers an easy to install and simple tool for running FreeSurfer workflows
- Using FSurf allows users to take advantage of the resources provided by OSG without having to become experts in running workflows on distributed systems

# More information

---

- [Documentation on Fsurf](#)
- [Github repo](#)
- [Freesurfer project page](#)

# Acknowledgements

---

- Don Krieger
- Rob Gardner
- Mats Rynge - help with Pegasus
- Lincoln Bryant - Infrastructure support