



# The NiPreps ecosystem for reproducible neuroimaging

Russ Poldrack  
Stanford University

# The standard design (anti-)pattern for fMRI preprocessing

- Pick a single software package
  - Usually based on considerations other than performance



# The standard design (anti-)pattern for fMRI preprocessing

- Pick a single software package
  - Usually based on considerations other than performance
- String together the tools from that package into a script to run the preprocessing workflow

```
#!/bin/bash

# Generate the subject list to make modifying this script
# to run just a subset of subjects easier.

for id in `seq -w 1 26` ; do
    subj="sub-$id"
    echo "====> Starting processing of $subj"
    echo
    cd $subj

    # If the brain mask doesn't exist, create it
    if [ ! -f anat/${subj}_T1w_brain_f02.nii.gz ] ; then
        echo "Skull-stripped brain not found, using bet with a fractional intensity thresh
# Note: This fractional intensity appears to work well for most of the subjects in
# Flanker dataset. You may want to change it if you modify this script for your own
bet2 anat/${subj}_T1w.nii.gz \
anat/${subj}_T1w_brain_f02.nii.gz -f 0.2
    fi

    # Copy the design files into the subject directory, and then
    # change "sub-08" to the current subject number
    cp ..design_run1.fsf .
    cp ..design_run2.fsf .

    # Note that we are using the | character to delimit the patterns
    # instead of the usual / character because there are / characters
    # in the pattern.
    sed -i '' "s|sub-08|${subj}|g" \
        design_run1.fsf
    sed -i '' "s|sub-08|${subj}|g" \
        design_run2.fsf

    # Now everything is set up to run feat
    echo "====> Starting feat for run 1"
    feat design_run1.fsf
    echo "====> Starting feat for run 2"
    feat design_run2.fsf
    echo

    # Go back to the directory containing all of the subjects, and repeat the loop
    cd ..
done
```

# The standard design (anti-)pattern for fMRI preprocessing

- Pick a single software package
  - Usually based on considerations other than performance
- String together the tools from that package into a script to run the preprocessing workflow
  - Generally written by a student or postdoc with little software engineering experience
- Hope it keeps working over time...
- Trust that it does the right thing...

```
#!/bin/bash

# Generate the subject list to make modifying this script
# to run just a subset of subjects easier.

for id in `seq -w 1 26` ; do
    subj="sub-$id"
    echo "====> Starting processing of $subj"
    echo
    cd $subj

    # If the brain mask doesn't exist, create it
    if [ ! -f anat/${subj}_T1w_brain_f02.nii.gz ] ; then
        echo "Skull-stripped brain not found, using bet with a fractional intensity thresh
# Note: This fractional intensity appears to work well for most of the subjects in
# Flanker dataset. You may want to change it if you modify this script for your own
bet2 anat/${subj}_T1w.nii.gz \
anat/${subj}_T1w_brain_f02.nii.gz -f 0.2
    fi

    # Copy the design files into the subject directory, and then
    # change "sub-08" to the current subject number
    cp ..//design_run1.fsf .
    cp ..//design_run2.fsf .

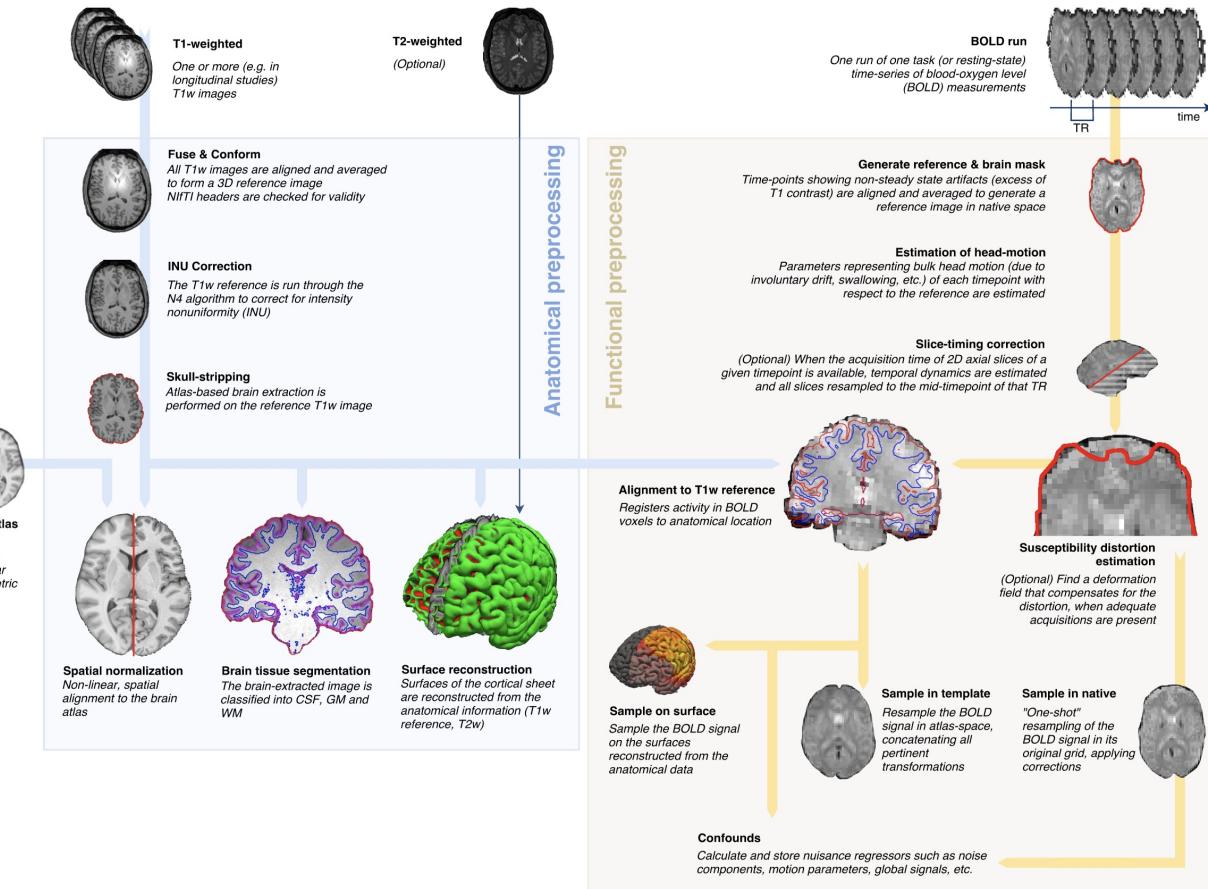
    # Note that we are using the | character to delimit the patterns
    # instead of the usual / character because there are / characters
    # in the pattern.
    sed -i '' "s|sub-08|${subj}|g" \
    design_run1.fsf
    sed -i '' "s|sub-08|${subj}|g" \
    design_run2.fsf

    # Now everything is set up to run feat
    echo "====> Starting feat for run 1"
    feat design_run1.fsf
    echo "====> Starting feat for run 2"
    feat design_run2.fsf
    echo

    # Go back to the directory containing all of the subjects, and repeat the loop
    cd ..
done
```

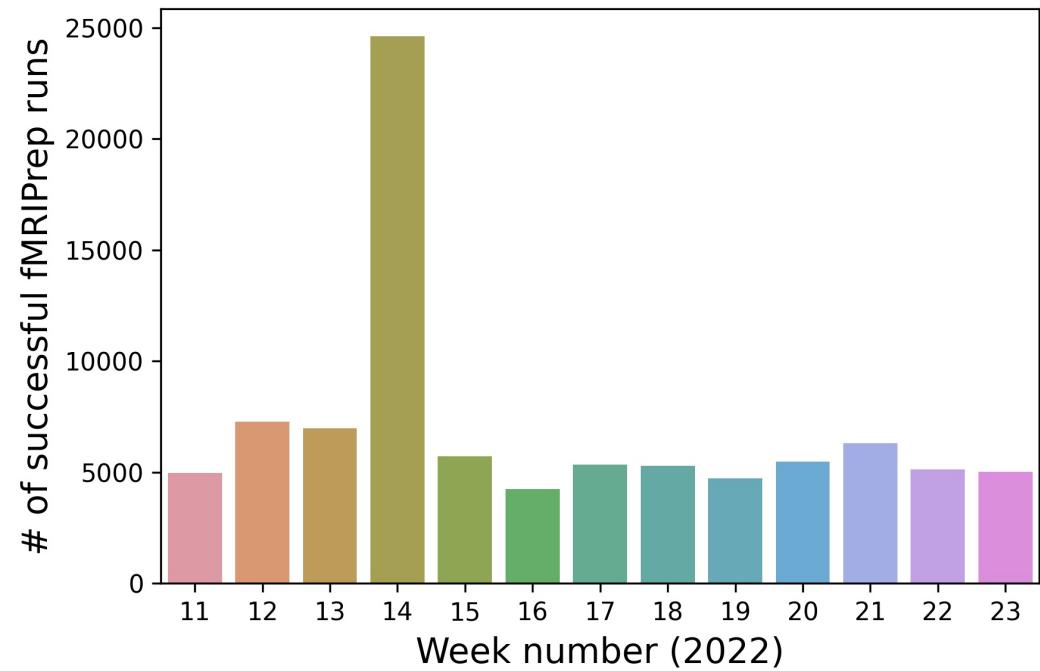
# Towards a new design pattern for fMRI preprocessing: fMRIPrep

- A robust workflow for preprocessing fMRI data
  - Tested on a large number of fMRI datasets from OpenNeuro
  - Adapts to each dataset to process it in the best way possible
  - Provides a common interface to a variety of tools
  - Provide visualization to help identify problems

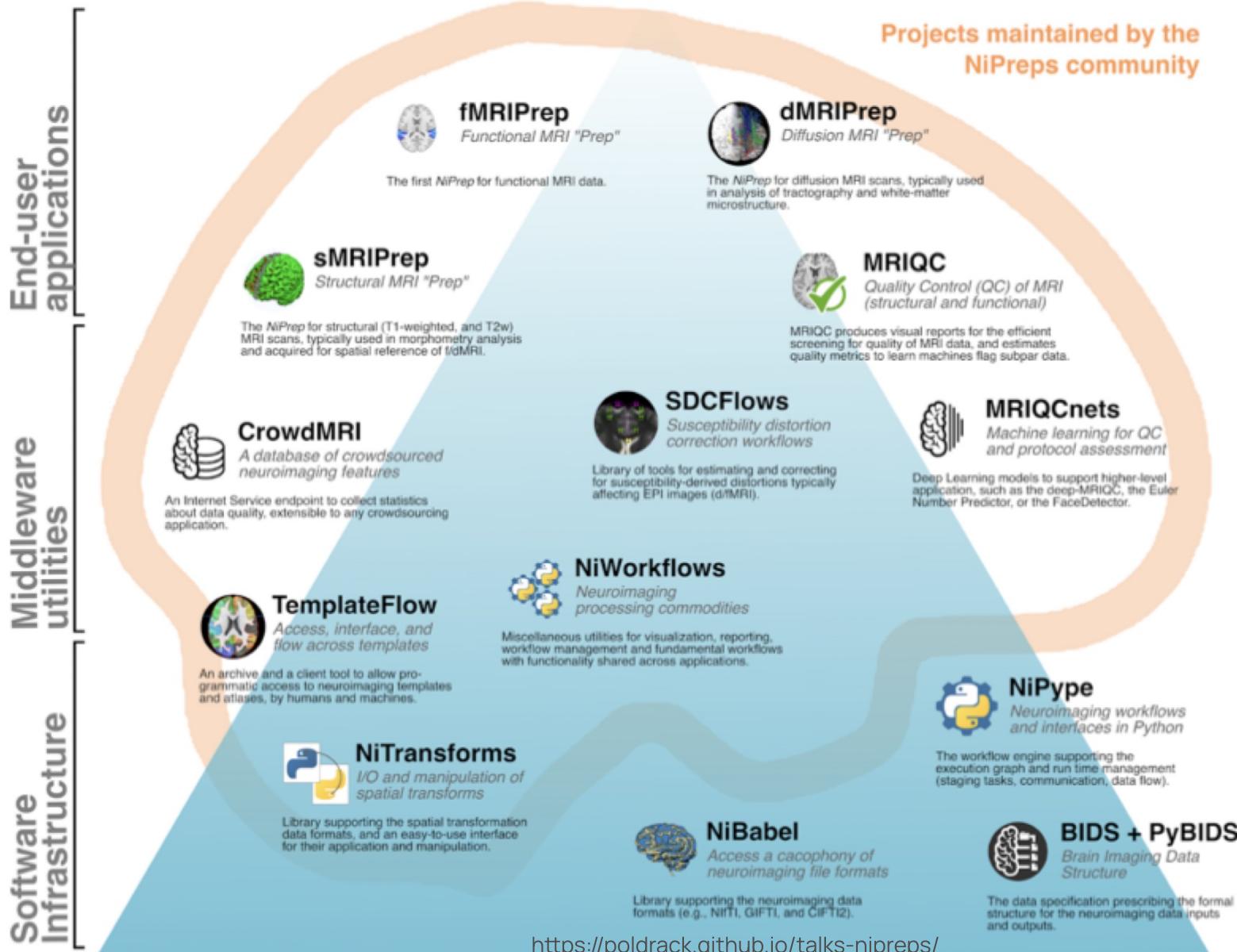


# fMRIPrep usage

- Usage tracked using an opt-out telemetry system
  - Allows quick identification of bugs and usage patterns



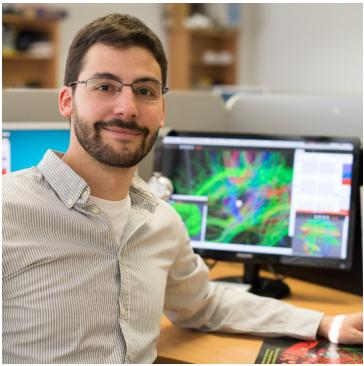
# NiPreps: An ecosystem of Neuroimaging Preprocessing Tools



# The NiPreps community



Poldracklab (Stanford)



Esteban Lab (CHUV Lausanne)



Satterthwaite Lab (Penn)  
Milham Lab (Child Mind Inst)



Ariel Rokem  
Research Associate Professor  
email: [arokem@uw.edu](mailto:arokem@uw.edu)  
[Website](#)



McKenzie Hagen  
Graduate student, started 2021



Ezgi Yucel  
Graduate student (with Ione Fine), started 2017



John Kruper  
Post-bacc student, started July 2020



Teresa Gomez  
Post-bacc student, started January 2022

Rokem Lab (UW)

# Nipreps principles

- **Openness and community engagement**
  - All work done in the open
  - Community-driven decision making

# Nipreps principles

- **Openness and community engagement**
  - All work done in the open
  - Community-driven decision making
- **Reproducibility and transparency**
  - Containerized applications following the BIDS-Apps specification
  - Detailed visual reporting
  - Citation boilerplate to ensure standards-compliant reporting

# Nipreps principles

- **Openness and community engagement**
  - All work done in the open
  - Community-driven decision making
- **Reproducibility and transparency**
  - Containerized applications following the BIDS-Apps specification
  - Detailed visual reporting
  - Citation boilerplate to ensure standards-compliant reporting
- **Generality, consistency, and robustness**
  - Fully based on the BIDS and BIDS-Derivatives standards
  - Tested using continuous integration

# Nipreps principles

- **Openness and community engagement**
  - All work done in the open
  - Community-driven decision making
- **Reproducibility and transparency**
  - Containerized applications following the BIDS-Apps specification
  - Detailed visual reporting
  - Citation boilerplate to ensure standards-compliant reporting
- **Generality, consistency, and robustness**
  - Fully based on the BIDS and BIDS-Derivatives standards
  - Tested using continuous integration
- **Modularity and longevity potential**
  - Reusable middleware components

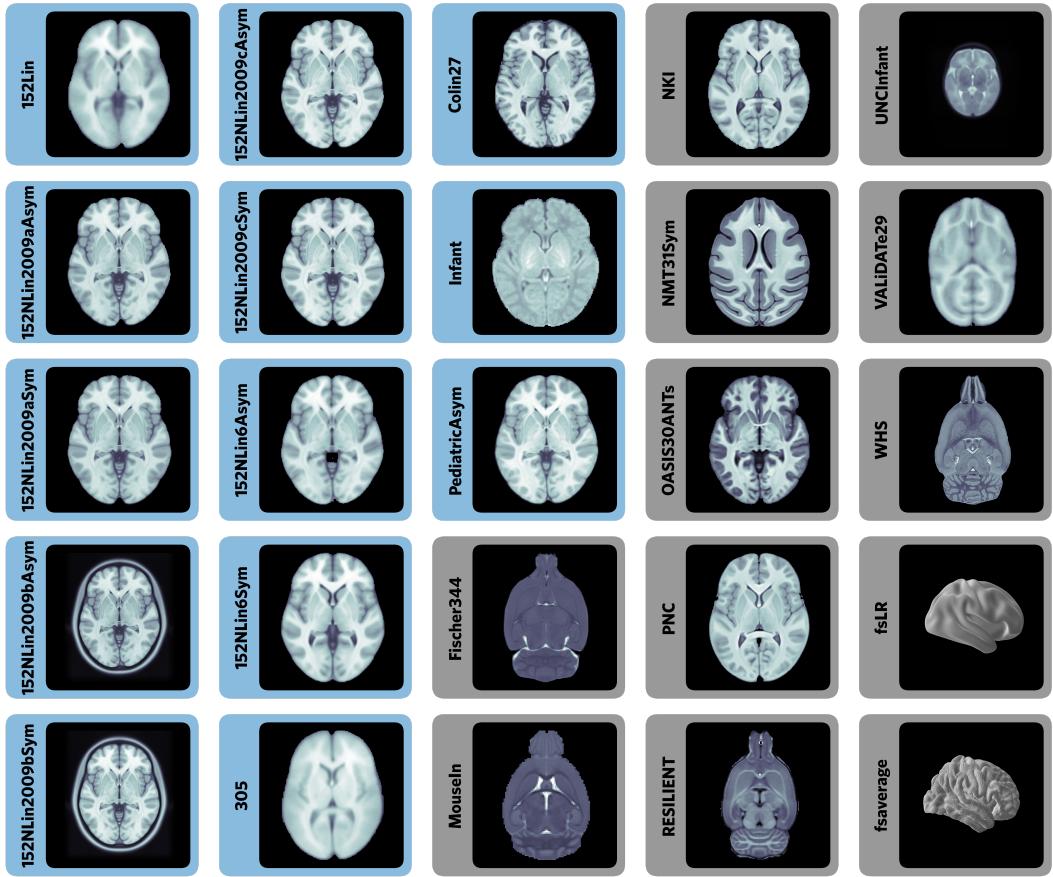
# RF1MH121867: NiPreps: integrating neuroimaging preprocessing workflows across modalities, populations, and species

- **Aim 1: Solidify and extend NiPreps infrastructure and middleware**
  - TemplateFlow: FAIR Sharing and management of neuroimaging templates and atlases
  - SDCflows: Integrating susceptibility distortion correction
  - NiTransforms: Integrating spatial transforms

# Middleware: TemplateFlow

- Templates and atlases are commonly used in neuroimaging research
- There is significant lack of clarity in the use of these templates
  - There are numerous versions of the widely used “MNI template”
- Templateflow provides programmatic access to a database of templates and mappings between them
- Python API client:

```
>>> from templateflow import api as tf
>>> tf.get("MNI152NLin2009cAsym",
...           suffix="T1w")
```



# Middleware: SDCFlows

- Susceptibility distortion correction is a significant source of analytical flexibility
  - Multiple fieldmap sequences (spin echo, gradient echo, scanner-reconstructed  $B_0$  maps)
  - Multiple correction implementations with different data structures
- Comparability among methods is poor, with little insight into sources of variation
- SDCFlows
  - Defines an intermediate representation of a field inhomogeneity map
  - Wraps or implements methods for estimating this map from different sequences
  - Provides a single method for applying estimated maps to EPI sequences
- SDCFlows is used in fMRIprep and dMRIprep, with plans to make a standalone tool for custom processing or easier interoperation with other pipelines

# Infrastructure: Architectural redesign

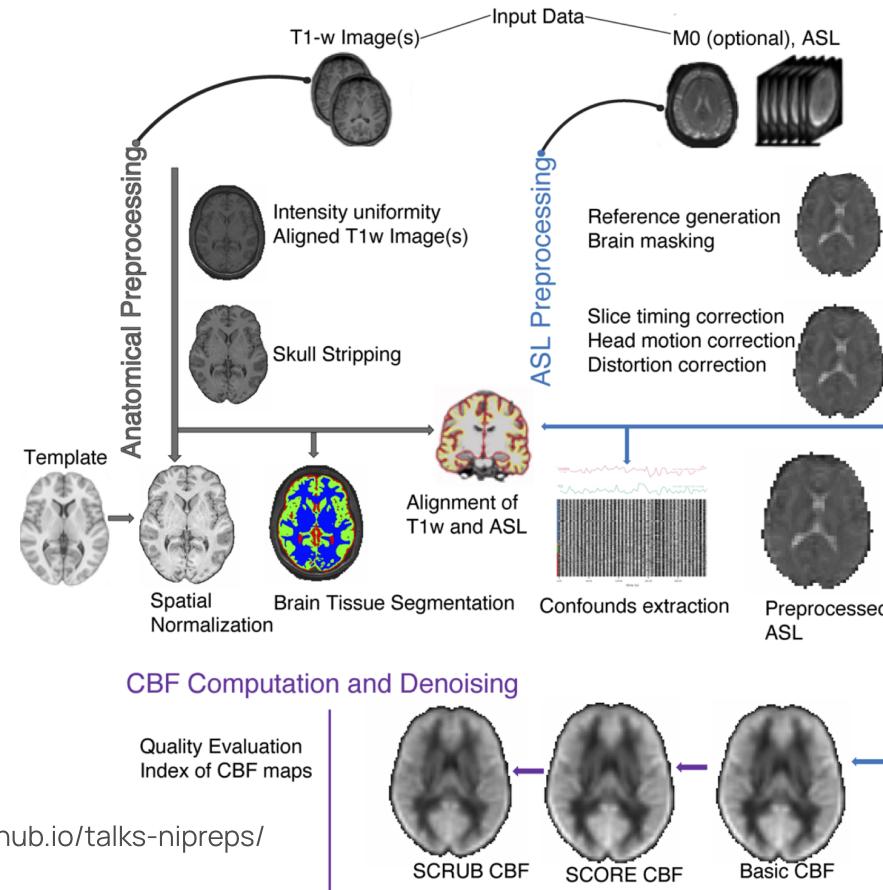
- Problem: fMRIPrep's "one size fits all" design has limitations for emerging use cases
  - Archiving preprocessing results requires balancing storage costs against possible use cases.
  - Including alternative algorithms requires custom code to integrate.
- Solution: Accept pre-computed derivatives and defer computationally cheap operations
  - E.g., Deep learning segmentations and masks can be accepted, skipping fMRIPrep defaults.
  - Multiple template registrations can be archived, analysts may resample BOLD series to different spaces on demand.
- This approach is implemented in SDCFlows and is being generalized to other components.

# RF1MH121867: NiPreps: integrating neuroimaging preprocessing workflows across modalities, populations, and species

- Aim 2: Expand the portfolio of end-user NiPreps
  - ASLPrep
  - dMRIPrep
  - PETPrep
  - NiBabies
  - NiRodents

# Workflows: ASLPrep (cerebral blood flow quantification)

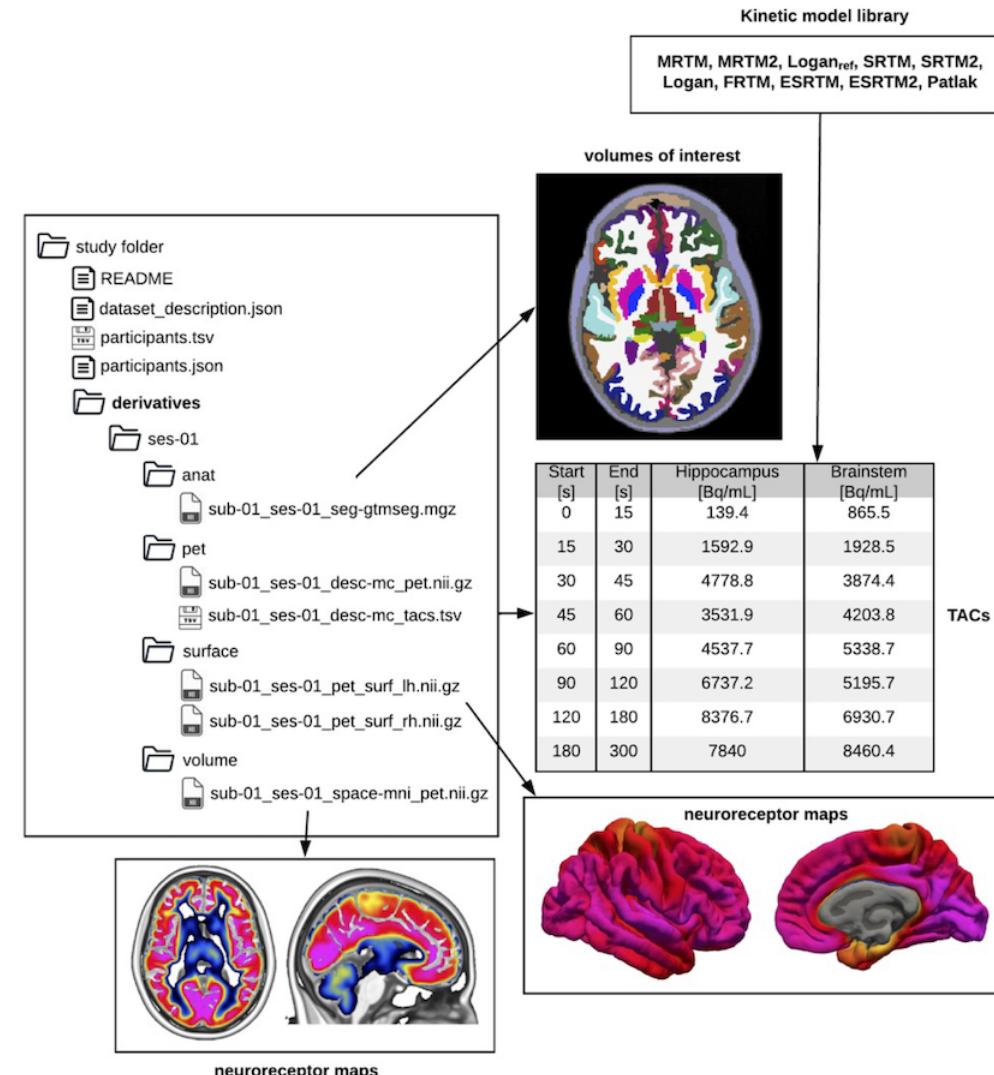
- A robust workflow for preprocessing arterial spin labeling (ASL) data
  - Including cerebral blood flow (CBF) quantification
  - Provides quality evaluation for CBF maps
  - Provides CBF quantification at the regional level using atlases



# Workflows: dMRIPrep (diffusion MRI)

# Workflows: PETPrep (positron emission tomography)

- A NiPreps workflow for PET preprocessing
  - Successfully merged petsurfer into nipype (1.8.0)
  - Incorporated nipype implementation of a robust head motion correction workflow (petprep\_hmc)
  - Developing a BIDS-Derivatives standard for PET derivatives



# Workflows: NiBabies (infant human MRI)

- Collaboraration with Damien Fair & HBCD team

image here

- **New developments**

- Support for pre-computed derivatives (mask, segmentations).
- Improved robustness and validity of CIFTI-2 outputs.

- **Upcoming developments**

- Morphometric outputs (cortical thickness, curvature)
- Improvements to susceptibility distortion correction versatility
- T2 assisted surface generation

# Workflows: NiRodents (rodent MRI)



# RF1MH121867: NiPreps: integrating neuroimaging preprocessing workflows across modalities, populations, and species

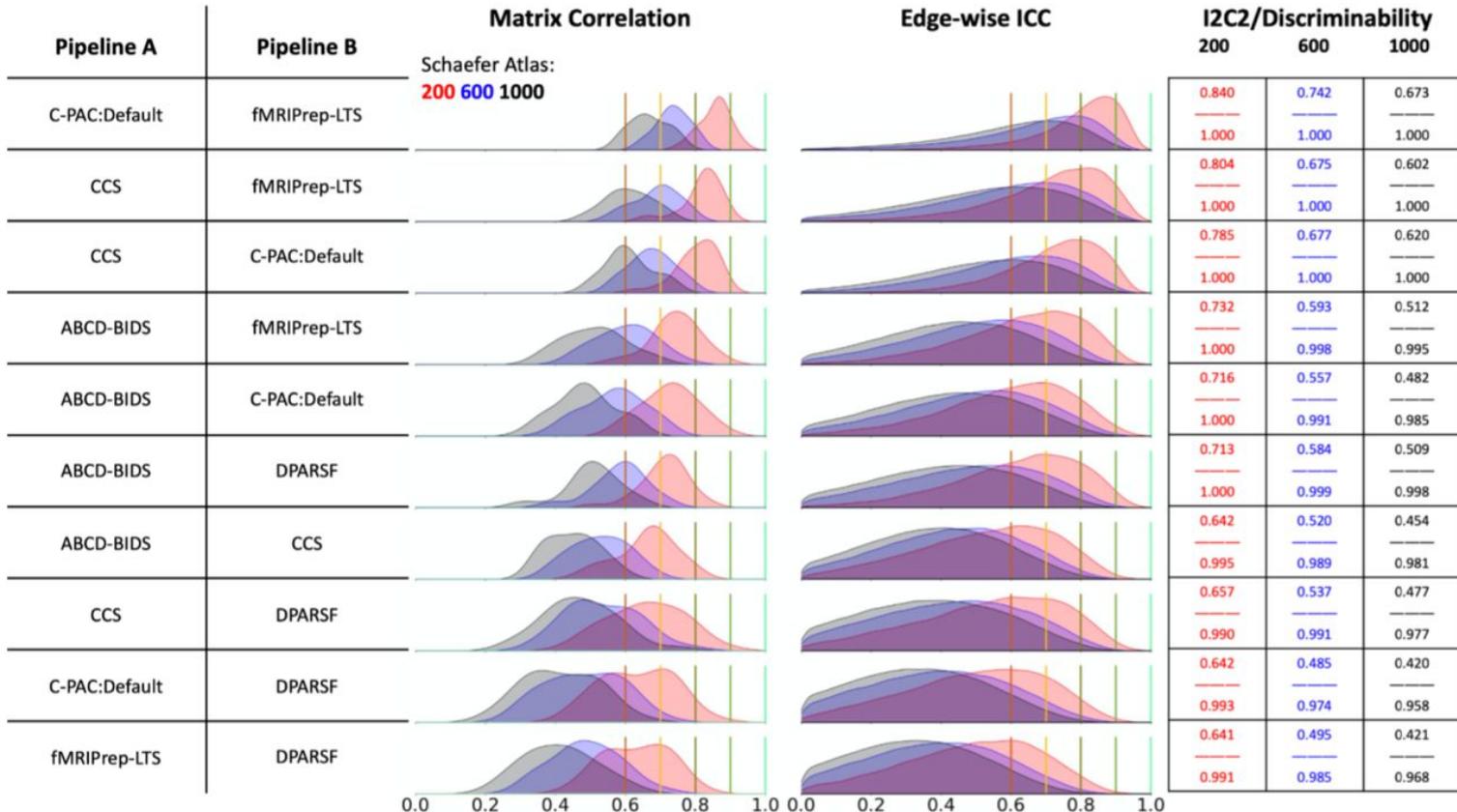
- **Aim 3: Consolidate the NiPreps community**
  - Project monitoring infrastructure: MIGAS
  - Evaluation of cross-workflow reproducibility
  - Hackathons and documentation

# Project monitoring: MIGAS

- An open-source, customizable telemetry solution
- Allows collecting usage information, errors, and status throughout a process's lifetime
- Easy to deploy with various cloud providers (Heroku / GCP / AWS)
- Available as a Python package:  
<https://pypi.org/project/migas/>

# Reproducibility: Cross-workflow evaluation

- CMI team developed a CPAC implementation of fMRIPrep
  - Able to achieve high levels of reproducibility in connectivity metrics between harmonized workflows
  - Helped identify causes of divergence, such as use of different versions of MNI template



# Hackathons and documentation

- Held a hackathon/documentation sprint in Glasgow following OHBM 2022
- Will participate in Brainhack Global 2022
- Planning to hold a hackathon/documentation sprint in Montreal in association with OHBM 2023

# Thank you!