

Introduction to MRIQC & fMRIPrep

Mitchell Horn
mhorn@bu.edu

Research Computing Services
Information Services & Technology
Boston University

Resources

- Tutorial Slides available:
 - <http://rcs.bu.edu/examples/imaging>
- Questions, Problems, Need Help?
 - help@scc.bu.edu
 - mhorn@bu.edu

Outline

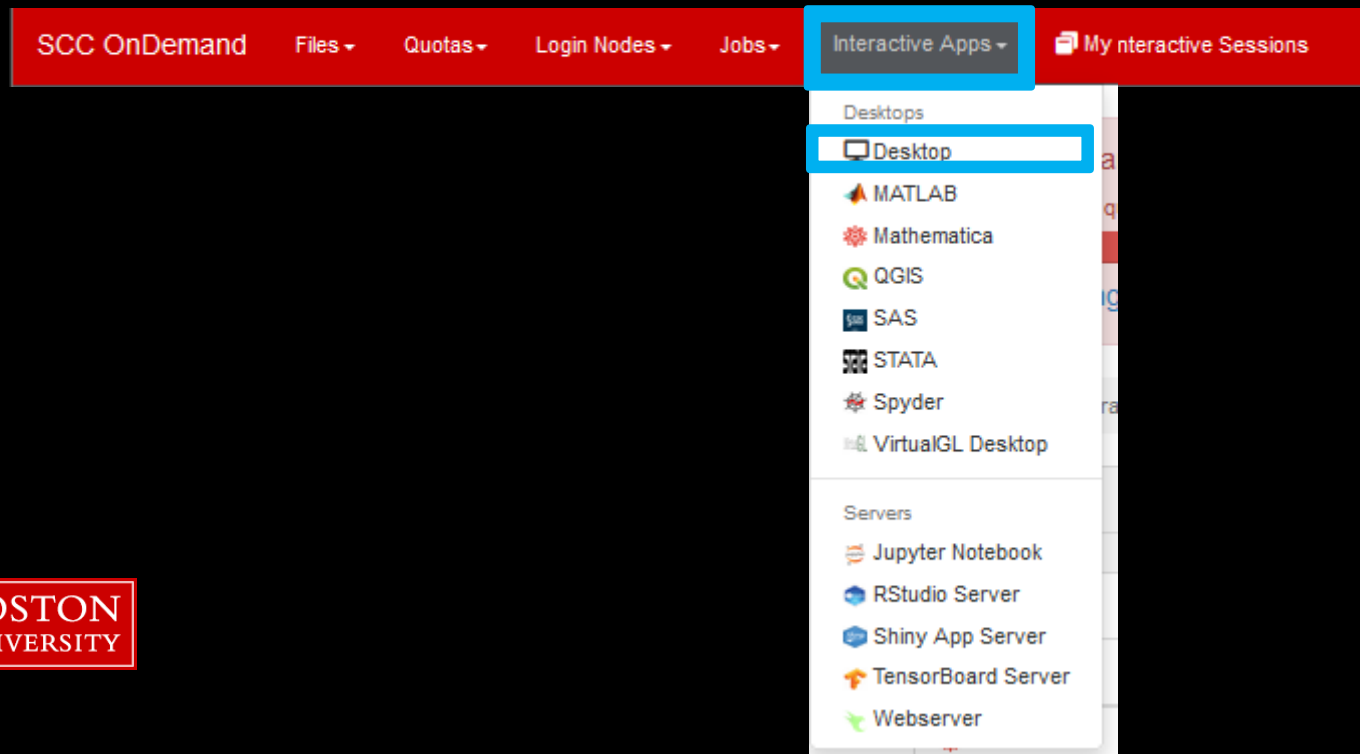
- Tutorial Setup (Hands-On)
- BIDS
- MRIQC
- fMRIPrep

Outline

- Tutorial Setup (Hands-On)
- BIDS
- MRIQC
- fMRIPrep

Open an OnDemand session

1. Go to: scc-ondemand.bu.edu
2. Interactive Apps
3. Desktop



Anybody Without an SCC Account?

- `scc-ondemand-tutorial.bu.edu`
- username: `tuta#`
- password:

Open an OnDemand session

Desktop
This app will launch an interactive desktop on a compute node.

List of modules to load (space separated)
 Select Modules

Working Directory
 Select Directory
The directory to start in. (Defaults to home directory.)

Initial command to run

Number of hours

Number of cores

Number of gpus

Project

Extra qsub options

I would like to receive an email when the session starts

* The Desktop session data for this session can be accessed under the data root directory.

12 hours

1 core

project-ID

click launch!

7

Open an OnDemand session

Desktop (6994379) 1 core | Running

Host: [>_scc-bb3](#) Delete

Created at: 2022-08-29 11:12:24 EDT

Time Remaining: 19 hours and 53 minutes

Session ID: [afff80fb-ca1f-44fd-a440-0637da849e84](#)

Compression Image Quality

0 (low) to 9 (high) 0 (low) to 9 (high)

[Connect to Desktop](#) View Only (Share-able Link)

click Connect to Desktop!

Open Tutorial File

```
[m]$ module load sublime
```

```
[m]$ sublime /project/scv/examples/imaging/tut_fmriprep_scc/notes.txt
```

Instruction #1

1. Copy the tutorial data to your local project space.

a. To see the SCC projects you're a part of:

```
[]$ groups
```

b. Copy the tutorial .zip file to your new directory:

```
[]$ cd /projectnb/YOUR_PROJECT/
```

```
[]$ cp /project/scv/examples/imaging/tut_fmripred_scc/fmripred_tutorial.zip .
```

```
[]$ unzip fmripred_tutorial.zip
```

```
[]$ cd fmripred_tutorial
```

Outline

- Tutorial Setup (Hands-On)
- **BIDS**
- MRIQC
- fMRIPrep

BIDS

- Brain Imaging Data Structure

Why?

- To standardize organization of complex data
- Have software develop BIDS apps to take in datasets
- Avoid lab-specific pipelines
- Facilitate validating datasets and curation
- Facilitate data sharing

BIDS


- Brain Imaging Data Structure

What?

- Three main file types in a BIDS dataset:
 - .JSON – contain metadata as key:value pairs
 - .TSV – contain tables of metadata
 - .NII.GZ – raw data files for fMRI and MRI data

Instruction #2

BIDS

- abcd/ 
- top level directory
 - contains the entire imaging study

BIDS

abcd/

└── sourcedata →

- level-1 subdirectory
- contains source imaging (.dcm, unorganized nifti, etc.)

BIDS

```
abcd/  
└── sourcedata  
    ├── 20180914133551640_191_S727038_I1048378.dcm  
    ├── 20180914133551640_191_S727038_I1048378.dcm  
    ├── 20180914133551640_191_S727038_I1048378.dcm  
    ├── 20180914133551640_191_S727038_I1048378.dcm  
    ├── 20180914133551640_191_S727038_I1048378.dcm  
    └── ...
```



• raw data

BIDS

abcd/

└─ sourcedata

└─ <DICOMS go here>

└─ sub-101



- level-1 subdirectory
- contains all imaging data in BIDS format

BIDS

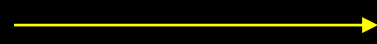
abcd/

└─ sourcedata

└─ <DICOMS go here>

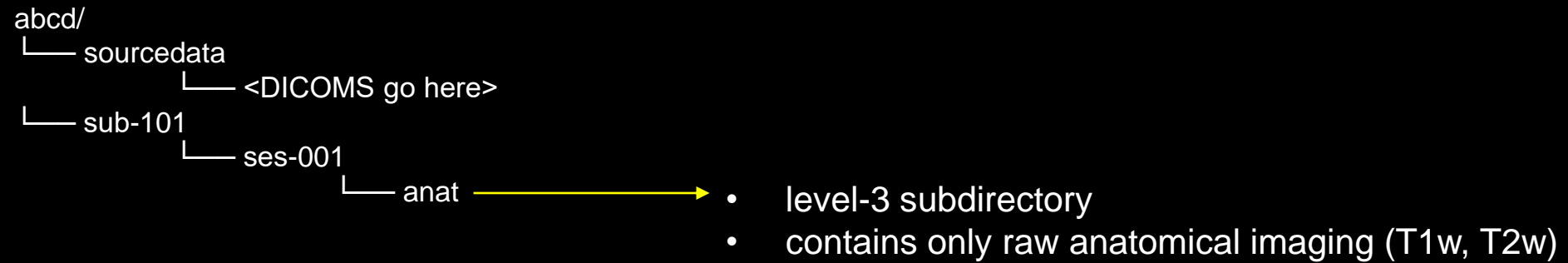
└─ sub-101

└─ ses-001

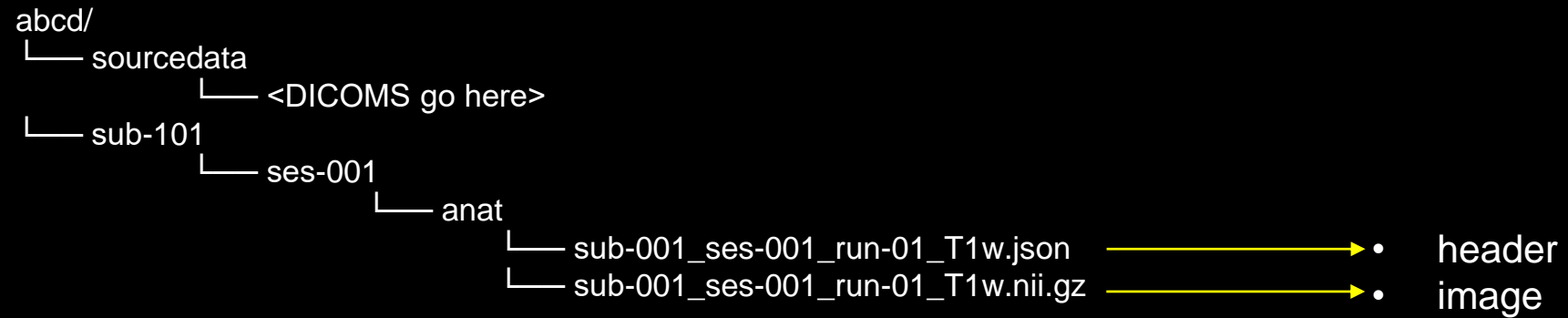


- level-2 subdirectory
- contains all BIDS imaging data from timepoint 001

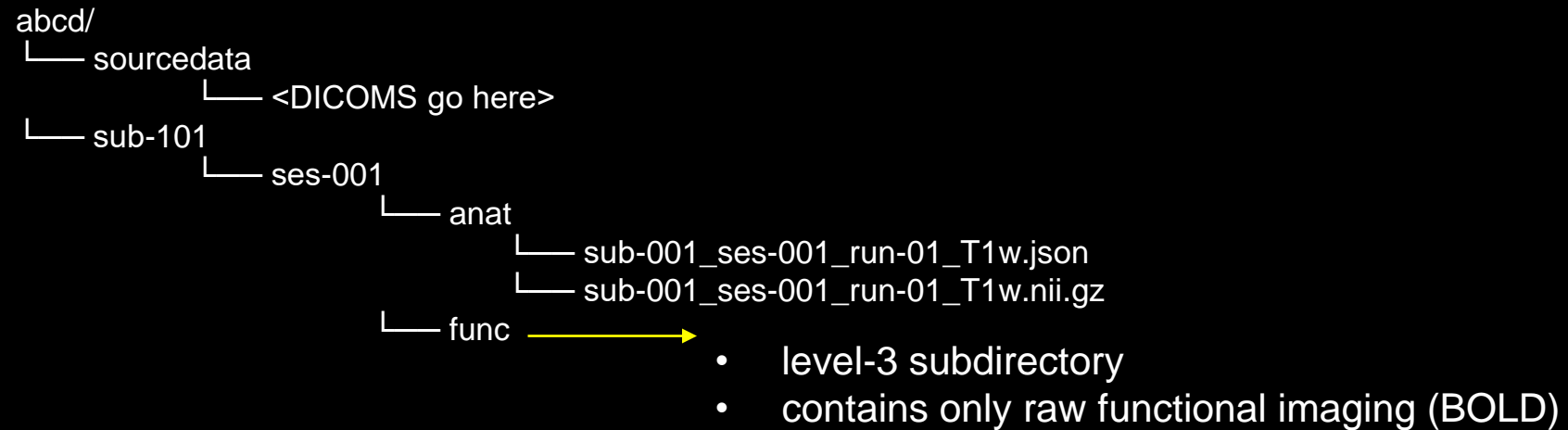
BIDS



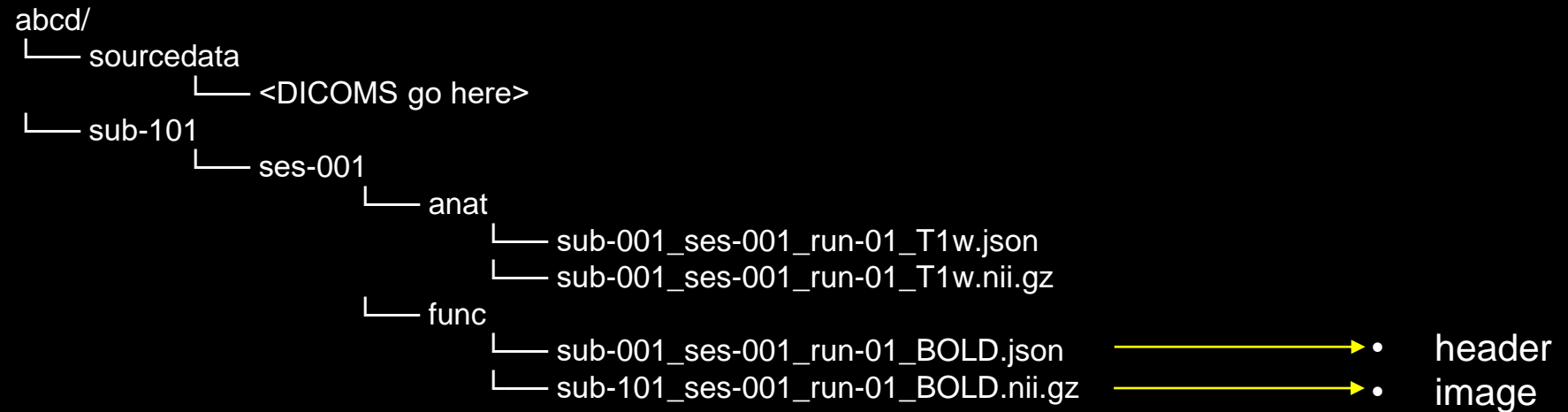
BIDS



BIDS



BIDS



BIDS

```
abcd/
├── sourcedata
│   └── <DICOMS go here>
├── sub-101
│   └── ses-001
│       ├── anat
│       │   ├── sub-001_ses-001_run-01_T1w.json
│       │   └── sub-001_ses-001_run-01_T1w.nii.gz
│       └── func
│           ├── sub-001_ses-001_run-01_BOLD.json
│           └── sub-101_ses-001_run-01_BOLD.nii.gz
```

README
dataset_description.json
participants.tsv

- text file describing the nature of your study
- information about the BIDS version, authors, licensing, etc.
- describes subject characteristics like age, sex, handedness etc.

BIDS

- Unpacking DICOMs
 - Reads through DICOM metadata to sort and stack slices/volumes into specified file format
- dcm2niix
 - MRICroGL
- **YAXIL - ArcGet.py**
- heudiconv
- dcmunpack / ununpacksdcmdir
 - FreeSurfer
- spm_dicom_convert
 - SPM
- to3d
 - AFNI

BIDS Compliance

- For BIDS apps to successfully recognize and import data, you can verify organization
- Not always necessary
- <http://bids-standard.github.io/bids-validator/>

Outline

- Tutorial Setup (Hands-On)
- BIDS
- **MRIQC**
- fMRIPrep

What is MRIQC?

- A software tool for extracting quality measures/performing quality assessments on T1w, T2w, and functional MRI data
- Fully automated pipeline assesses data quality and enables visual examination of MRI scans
- Outputs reports assessing Image Quality Metrics (IQMs) for each subject
 - Numerical output, can compare across subjects
- Runs as a singularity (like Heudiconv) on CCV
- Built on established software toolboxes (FSL, ANT's, and AFNI)
- Developed by the Poldrack Lab at Stanford University

Features of MRIQC

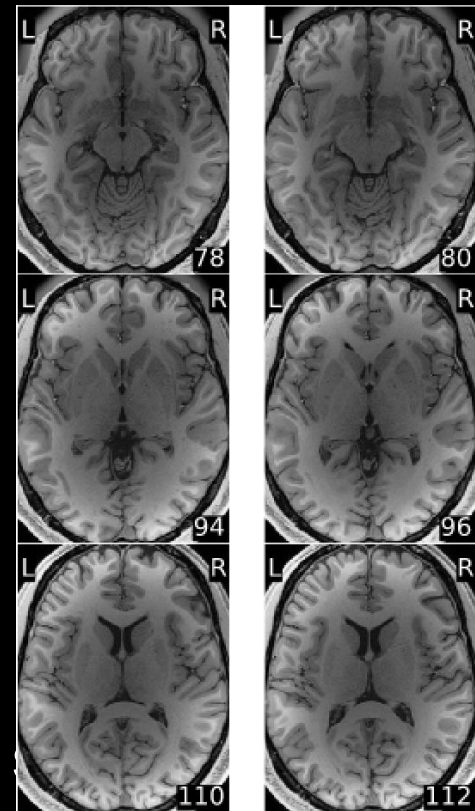
- Containerized versions bundle software dependencies together for easy installation on local servers and workstations
 - Can also be run through a free online service (OpenNeuro.org)
- No-reference quality norms help quantify quality differences across study sites
- Brain Imaging Data Structure (BIDS) compatible
- Easy integration with popular pipelines (fMRIPrep; NiPy)
- Automatically generates anonymized HTML reports
 - Can easily share the summarized QA data between group members and collaborators over email or other low-security apps

How does MRIQC improve QA?

- Problems with visual QA
 - Subjective
 - Impractical for large data sets
 - Time consuming
 - Prone to variability (inter-rater differences)
 - Some artifacts evade human detection
- Help with large scale imaging efforts
 - Less subject to site or population-specific biases
 - Standardize procedures between sites

MRIQC Mechanisms

- Image Quality Metrics (IQM)
 - No-reference metric of data quality
- Structural IQM Categories
 - Signal and noise comparisons
 - Information theory measures
 - Mechanical artifacts
- Functional IQM Categories
 - Spatial artifact detection
 - Temporal stability
- For more information on IQM's please the tutorial MRIQC PT2: IQMS



Other relevant info: What are Artifacts?

- A feature appearing in an image that is not in the original object
- May cause unwanted distortions in brain images
- Do not always affect image quality but can affect diagnostic quality
 - i.e. an artifact may be confused with pathology
- Some artifacts cannot be detected by the human eye - programs like MRIQC help to detect these
- Two main overarching categories:
 - Machine specific artifacts
 - Patient related artifacts
- See the tutorial titled MRIQC PT3: Artifacts for more information

MRIQC Doc

<https://mriqc.readthedocs.io/en/latest/index.html>

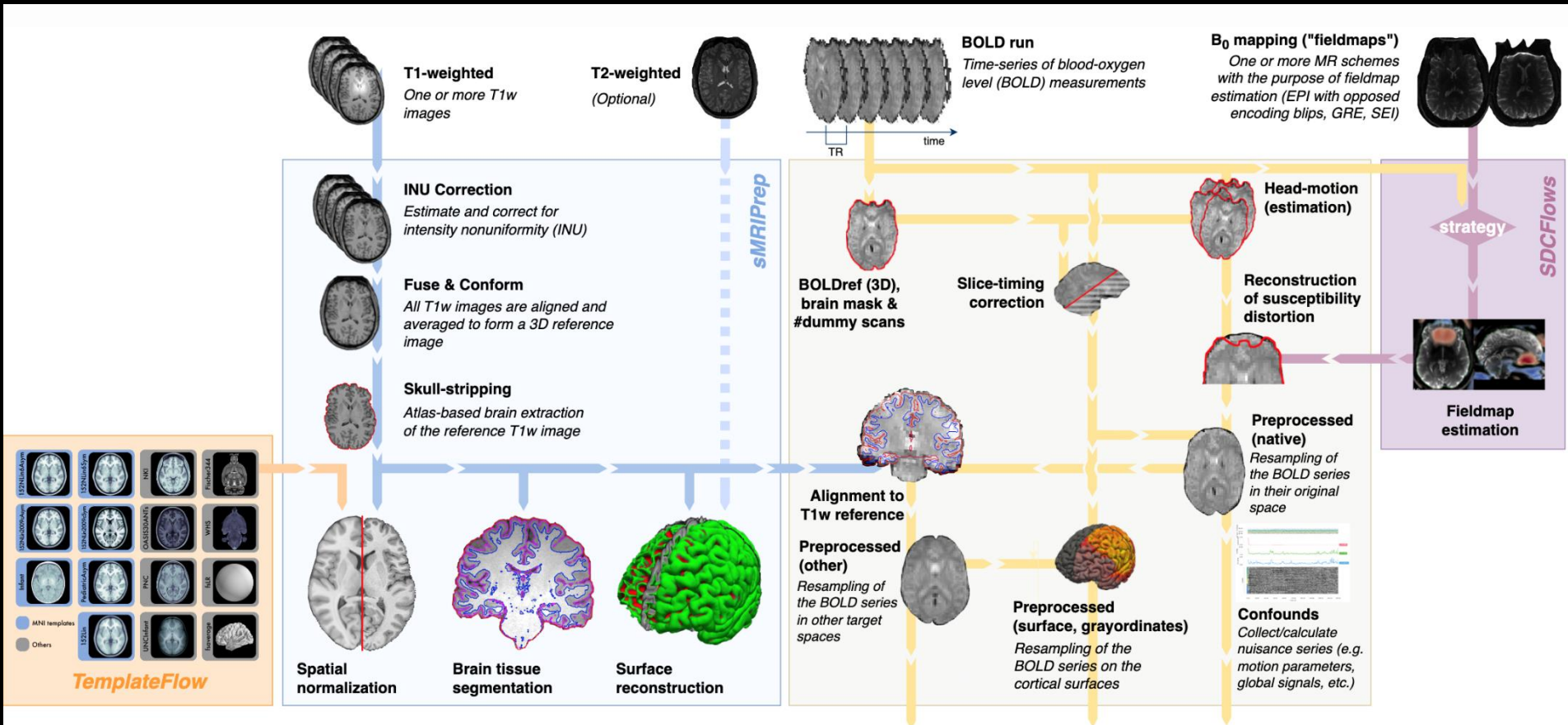
Outline

- Tutorial Setup (Hands-On)
- BIDS
- MRIQC
- fMRIPrep

What's fMRIPrep

- fMRI data preprocessing pipeline
- provides state-of-the-art interface
- provides robust support to variations of scans
- provides easy interface for inspection
- provides outputs ready for group analysis
- requires minimal input

What's fMRIPrep



Why To Not Use fMRIPrep

1. Very narrow FoV images oftentimes do not contain enough information for standard image registration methods to work correctly.
2. *fMRIPrep* may also underperform for particular populations (e.g., infants) and non-human brains, although appropriate templates can be provided to overcome the issue.
3. The “EPInorm” approach is currently not supported, although we plan to implement this feature (see [#620](#)).
4. If you really want unlimited flexibility (which is obviously a double-edged sword).
5. If you want students to suffer through implementing each step for didactic purposes, or to learn shell-scripting or Python along the way.
6. If you are trying to reproduce some *in-house* lab pipeline

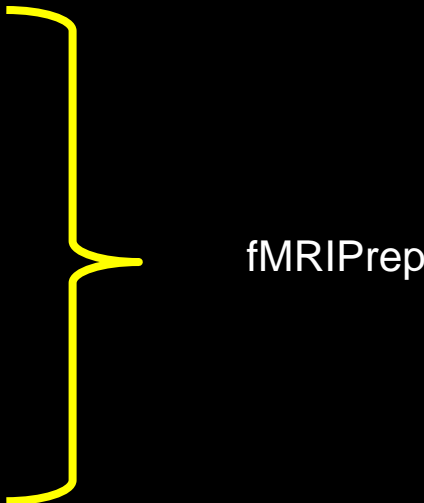
Outline

- Tutorial Setup (Hands-On)
- BIDS
- XNAT & YAXIL
- fMRIPrep

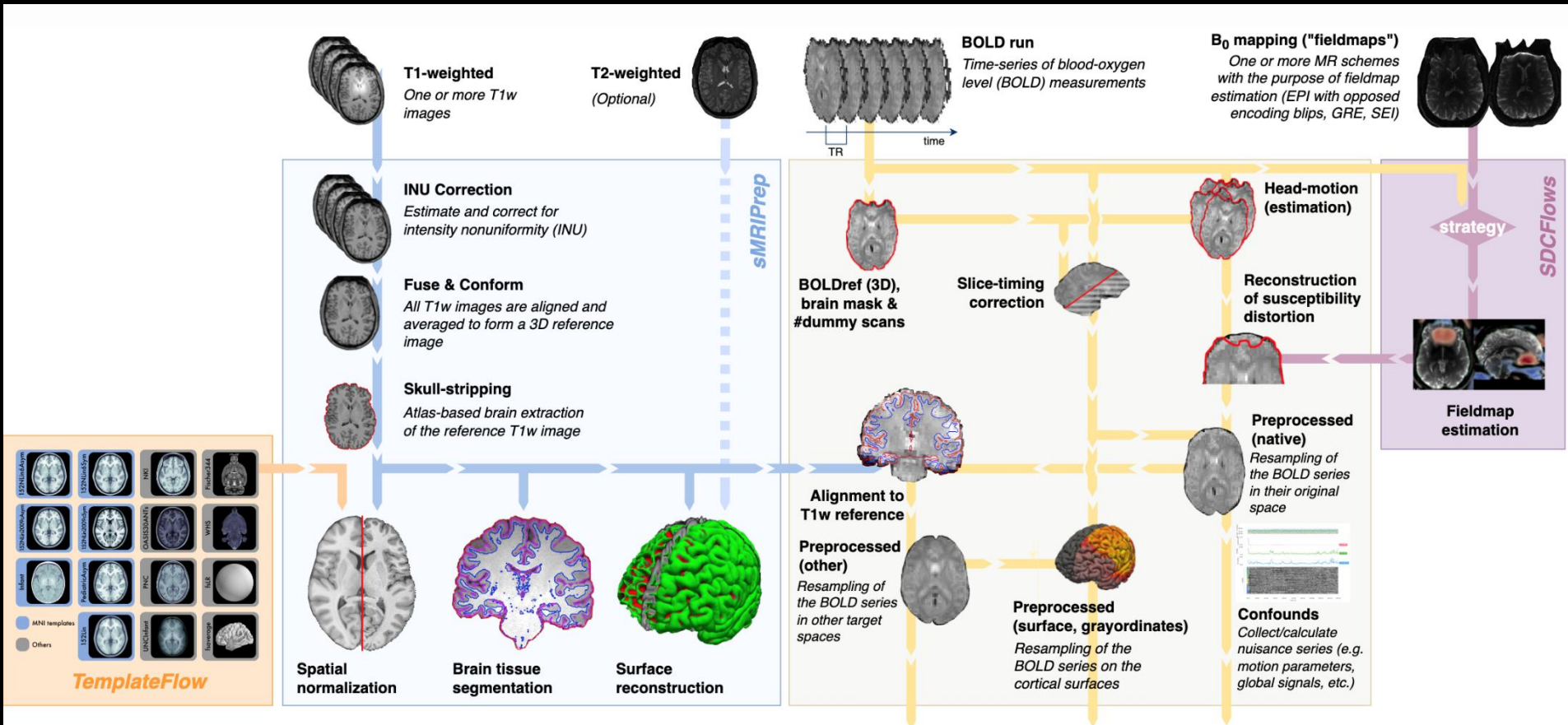
fMRIPrep

- Workflow taking principal input images and generates a standardized preprocessed output for analysis
- Uses BIDS standard for input and output
- Customize preprocessing via one command line with arguments
- Generate visual outputs for basic QC analysis

fMRIPrep

- Adapts multiple software packages into a single standardized process.
 - FreeSurfer
 - FSL
 - AFNI
 - ANTS
 - Connectomewb
 - ICA-AROMA
 - Python
- 
- fMRIPrep

fMRIPrep



fMRIPrep

```
fmriprep BIDS BIDS/derivatives participant [options]
```



command to run fmriprep

fMRIPrep

```
fmriprep BIDS BIDS/derivatives participant [options]
```



input directory

fMRIPrep

```
fmriprep BIDS BIDS/derivatives participant [options]
```



output directory

fMRIPrep

```
fmriprep BIDS BIDS/derivatives participant [options]
```



analysis level

fMRIPrep

fmriprep BIDS BIDS/derivatives participant [options]



tons of options!

fMRIPrep

■ Some Useful Options

■ Performance control

- `--nprocs` : number of cpus for entire process
- `--omp-nthreads` : number of threads per process
- `--mem` : upper bound memory limit for entire process

■ Workflow configurations

- `--output-spaces` : list spaces i.e. (MNI251Lin6Asym, MNI152Lin2009cAsym)
- `--bold2t1w-init` : DOF when registering BOLD to T1w (6=def, options 6, 9, 12)
- `--dummy-scans` : number of nonsteady-state volumes
- `--output-spaces` : choose your standardized spaces

■ Estimating Confounds options

- `--use-aroma` : add ICA_AROMA to your processing steam
- `--fd-spike-threshold` : framewise displacement threshold for flagging frame as an outlier
- `--dvars-spike-threshold` : DVARS threshold for flagging frame as an outlier

■ FreeSurfer options

- `--fs-license-file` : path to FreeSurfer license key
- `--fs-subjects-dir` : path to existing FreeSurfer directory to import
- `--fs-no-reconall` : disable FreeSurfer preprocessing

fMRIPrep

```
fmriprep /projectnb/BIDS /projectnb/BIDS/derivatives/fmriprep
participant --nprocs 4 --omp-nthreads 4 --mem 16 --use-aroma --
output-spaces MNI152NLin2009cAsym:res-2 fsaverage --fd-spike-
threshold 0.5 --dvars-spike-threshold 1.5 --use-syn-sdc warn --fs-
license-file /share/pkg.7/freesurfer/7.3.2/install/license --fs-
subjects-dir /projectnb/BIDS/derivatives/freesurfer --skip-bids-
validation --participant-label $subjid
```


fMRIPrep

```
fmriprep /projectnb/BIDS /projectnb/BIDS/derivatives/fmriprep
participant -w /scratch/$subjid --nprocs 4 --omp-nthreads 4 --mem 16
--use-aroma --output-spaces MNI152NLin2009cAsym:res-2 fsaverage --fd-
spike-threshold 0.5 --dvars-spike-threshold 1.5 --use-syn-sdc warn --
fs-license-file /share/pkg.7/freesurfer/7.3.2/install/license --fs-
subjects-dir /projectnb/BIDS/derivatives/freesurfer --skip-bids-
validation --participant-label $subjid
```

- Your base command and arguments

fMRIPrep

```
fmriprep /projectnb/BIDS /projectnb/BIDS/derivatives/fmriprep
participant -w /scratch/$subjid --nprocs 4 --omp-nthreads 4 --mem 16
--use-aroma --output-spaces MNI152NLin2009cAsym:res-2 fsaverage --fd-
spike-threshold 0.5 --dvars-spike-threshold 1.5 --use-syn-sdc warn --
fs-license-file /share/pkg.7/freesurfer/7.3.2/install/license --fs-
subjects-dir /projectnb/BIDS/derivatives/freesurfer --skip-bids-
validation --participant-label $subjid
```

- `w` : specify the working directory
 - There is significant I/O, so putting it on the /scratch will facilitate use on the SCC. This is local to the node that is processing your job instead of your project space.

fMRIPrep

```
fmriprep /projectnb/BIDS /projectnb/BIDS/derivatives/fmriprep
participant -w /scratch/$subjid --nprocs 4 --omp-nthreads 4 --mem 16
--use-aroma --output-spaces MNI152NLin2009cAsym:res-2 fsaverage --fd-
spike-threshold 0.5 --dvars-spike-threshold 1.5 --use-syn-sdc warn --
fs-license-file /share/pkg.7/freesurfer/7.3.2/install/license --fs-
subjects-dir /projectnb/BIDS/derivatives/freesurfer --skip-bids-
validation --participant-label $subjid
```

- `nprocs` : restrict entire job to 4 cpus
- `mem` : restrict entire job to 16 GB memory
- `omp-nthreads` : restrict single process to 4 cpus

fMRIPrep

```
fmriprep /projectnb/BIDS /projectnb/BIDS/derivatives/fmriprep
participant -w /scratch/$subjid --nprocs 4 --omp-nthreads 4 --mem 16
--use-aroma --output-spaces MNI152NLin2009cAsym:res-2 fsaverage --fd-
spike-threshold 0.5 --dvars-spike-threshold 1.5 --use-syn-sdc warn --
fs-license-file /share/pkg.7/freesurfer/7.3.2/install/license --fs-
subjects-dir /projectnb/BIDS/derivatives/freesurfer --skip-bids-
validation --participant-label $subjid
```

- use-aroma : add ICA_AROMA to process stream

fMRIPrep

```
fmriprep /projectnb/BIDS /projectnb/BIDS/derivatives/fmriprep
participant -w /scratch/$subjid --nprocs 4 --omp-nthreads 4 --mem 16
--use-aroma --output-spaces MNI152NLin2009cAsym:res-2 fsaverage --fd-
spike-threshold 0.5 --dvars-spike-threshold 1.5 --use-syn-sdc warn --
fs-license-file /share/pkg.7/freesurfer/7.3.2/install/license --fs-
subjects-dir /projectnb/BIDS/derivatives/freesurfer --skip-bids-
validation --participant-label $subjid
```

- **output-spaces** : standardized spaces you want
 - specify common spaces like MNI, fsaverage, fsnative
 - specify resolution of spaces like 0.5mm, 2mm, 1mm
 - specify custom template spaces
 - <https://fmriprep.org/en/22.1.1/spaces.html>

fMRIPrep

```
fmriprep /projectnb/BIDS /projectnb/BIDS/derivatives/fmriprep
participant -w /scratch/$subjid --nprocs 4 --omp-nthreads 4 --mem 16
--use-aroma --output-spaces MNI152NLin2009cAsym:res-2 fsaverage --fd-
spike-threshold 0.5 --dvars-spike-threshold 1.5 --use-syn-sdc warn --
fs-license-file /share/pkg.7/freesurfer/7.3.2/install/license --fs-
subjects-dir /projectnb/BIDS/derivatives/freesurfer --skip-bids-
validation --participant-label $subjid
```

- `fd-spike-threshold` : set threshold for detection
- `dvar-spike-threshold` : set threshold for detection

fMRIPrep

```
fmriprep /projectnb/BIDS /projectnb/BIDS/derivatives/fmriprep
participant -w /scratch/$subjid --nprocs 4 --omp-nthreads 4 --mem 16
--use-aroma --output-spaces MNI152NLin2009cAsym:res-2 fsaverage --fd-
spike-threshold 0.5 --dvars-spike-threshold 1.5 --use-syn-sdc warn --
fs-license-file /share/pkg.7/freesurfer/7.3.2/install/license --fs-
subjects-dir /projectnb/BIDS/derivatives/freesurfer --skip-bids-
validation --participant-label $subjid
```

- `use-syn-sdc` : use ANTs' symmetric normalization
 - method of fieldmap-less distortion correction

fMRIPrep

```
fmriprep /projectnb/BIDS /projectnb/BIDS/derivatives/fmriprep
participant -w /scratch/$subjid --nprocs 4 --omp-nthreads 4 --mem 16
--use-aroma --output-spaces MNI152NLin2009cAsym:res-2 fsaverage --fd-
spike-threshold 0.5 --dvars-spike-threshold 1.5 --use-syn-sdc warn --
fs-license-file /share/pkg.7/freesurfer/7.3.2/install/license --fs-
subjects-dir /projectnb/BIDS/derivatives/freesurfer --skip-bids-
validation --participant-label $subjid
```

- `fs-license-file` : specify license (can use system)
- `fs-subjects-dir` : specify existing FreeSurfer output

fMRIPrep

```
fmriprep /projectnb/BIDS /projectnb/BIDS/derivatives/fmriprep
participant -w /scratch/$subjid --nprocs 4 --omp-nthreads 4 --mem 16
--use-aroma --output-spaces MNI152NLin2009cAsym:res-2 fsaverage --fd-
spike-threshold 0.5 --dvars-spike-threshold 1.5 --use-syn-sdc warn --
fs-license-file /share/pkg.7/freesurfer/7.3.2/install/license --fs-
subjects-dir /projectnb/BIDS/derivatives/freesurfer --skip-bids-
validation --participant-label $subjid
```

- skip-bids-validation : don't verify BIDS
 - save yourself the headache!

fMRIPrep

```
fmriprep /projectnb/BIDS /projectnb/BIDS/derivatives/fmriprep
participant -w /scratch/$subjid --nprocs 4 --omp-nthreads 4 --mem 16
--use-aroma --output-spaces MNI152NLin2009cAsym:res-2 fsaverage --fd-
spike-threshold 0.5 --dvars-spike-threshold 1.5 --use-syn-sdc warn --
fs-license-file /share/pkg.7/freesurfer/7.3.2/install/license --fs-
subjects-dir /projectnb/BIDS/derivatives/freesurfer --skip-bids-
validation --participant-label $subjid
```

- participant-label : specify your subject
 - should be in your BIDS directory!

Instruction #8 & #9

8. Review the simplified single subject submission file to s

```
[ ]$ cd qsub
```

```
[ ]$ sublime fmriprep_single.qsub
```

9. Make the changes required for submission:

a. Open the qsub file:

```
[ ]$ sublime fmriprep_single.qsub
```

b. Add your project to lines #4

```
4. #$ -P YOUR_PROJECT_CODE
```

c. Add the output directory to store the submission files

```
19. #$ -o /my/directory
```

d. Add the path to your input BIDS and output directories

Instruction #10 & #11

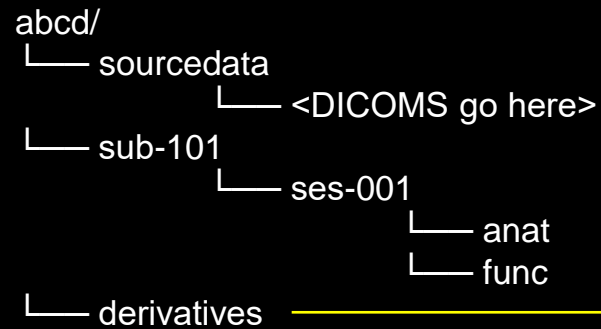
10. Submit the job

```
[ ]$ qsub fmriprep_single.qsub
```

11. Check that your submission starts (can take a few minutes):

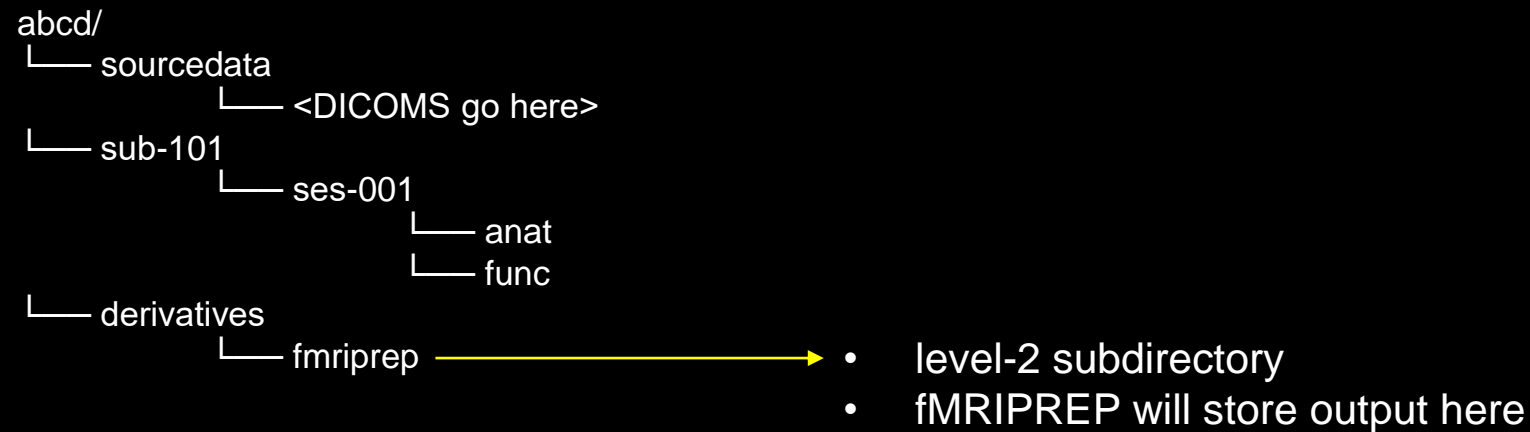
```
[ ]$ watch qstat -u YOUR_USERNAME
```

BIDS

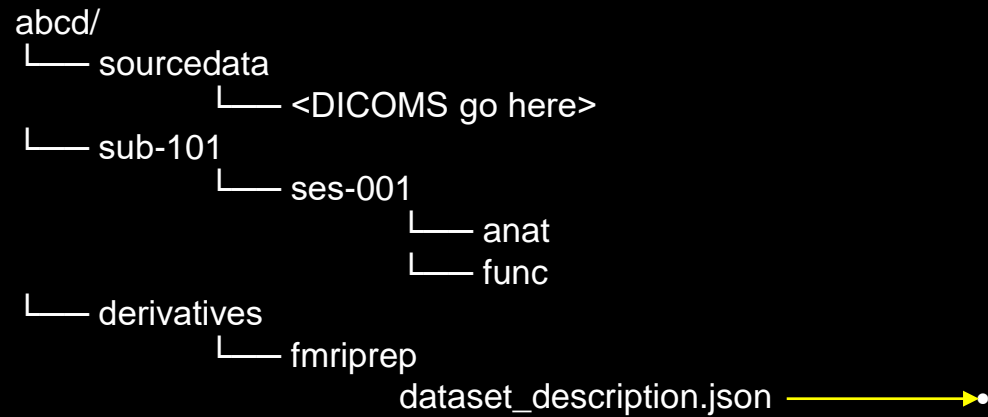


- level-1 subdirectory
- BIDS apps will store output here
- contains only derived imaging data

BIDS

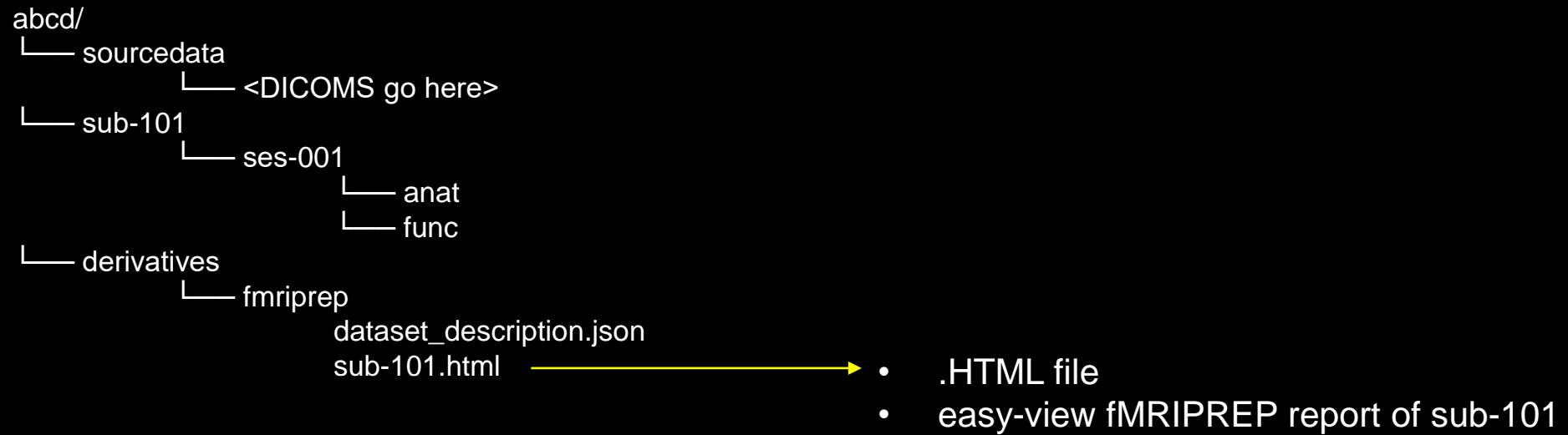


BIDS

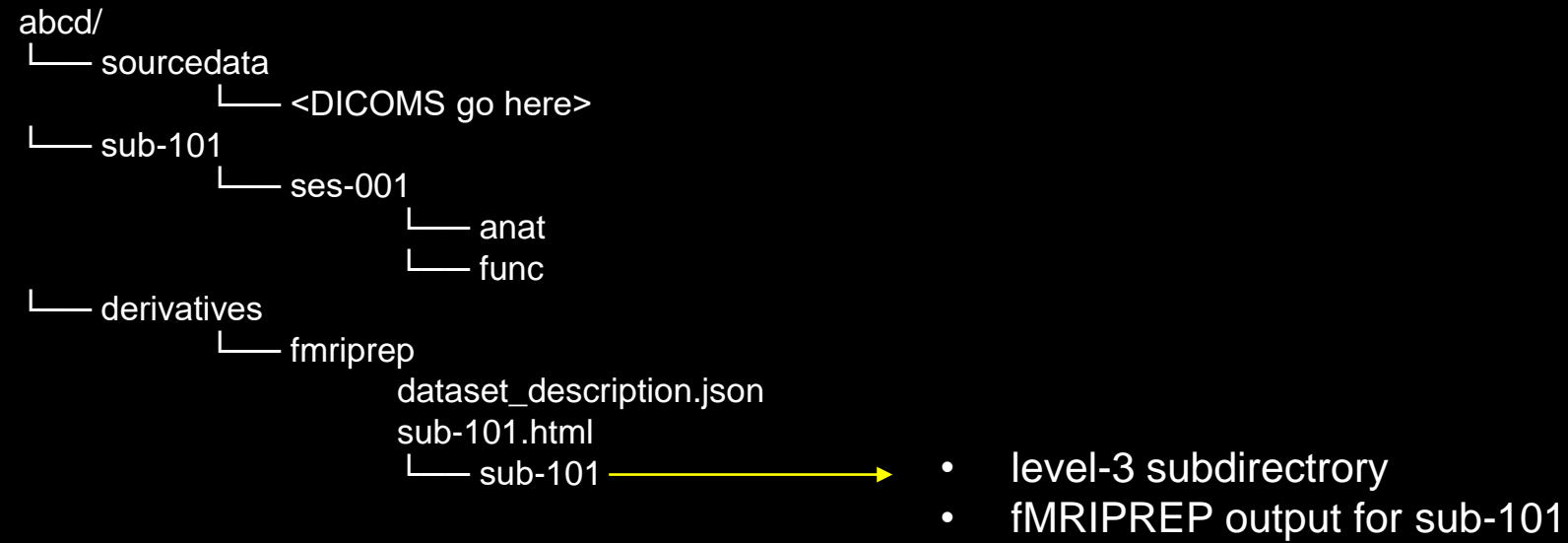


- JSON file
- Information about dataset
 - fmriprep version, authors, funding,

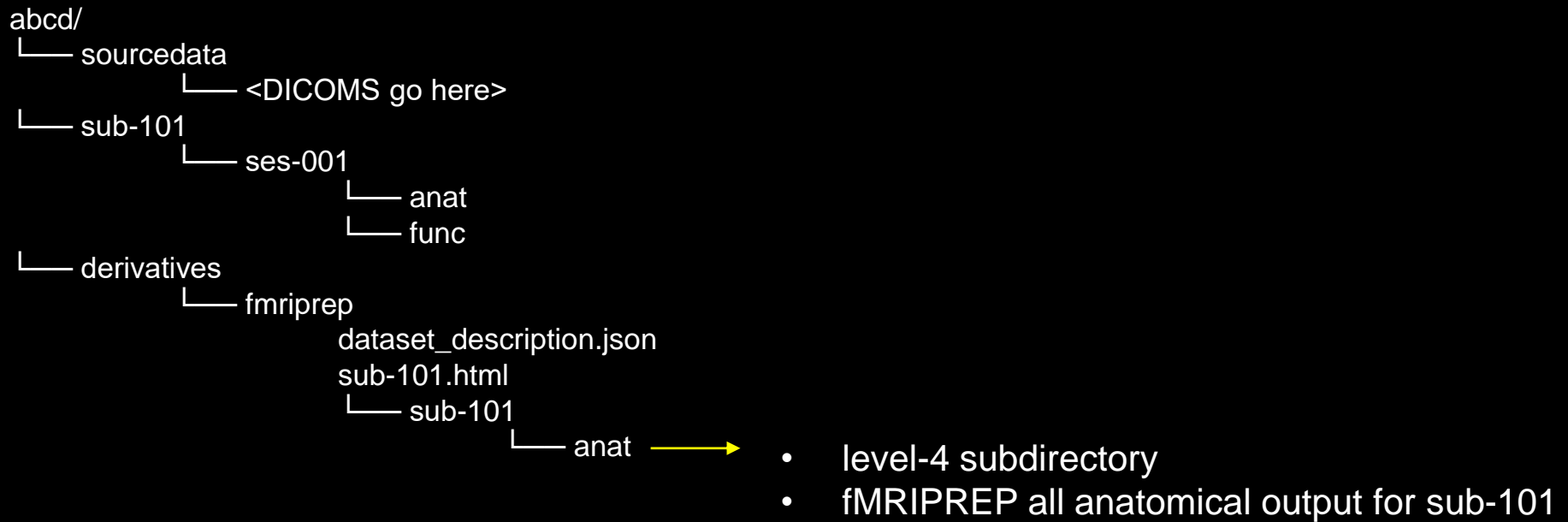
BIDS



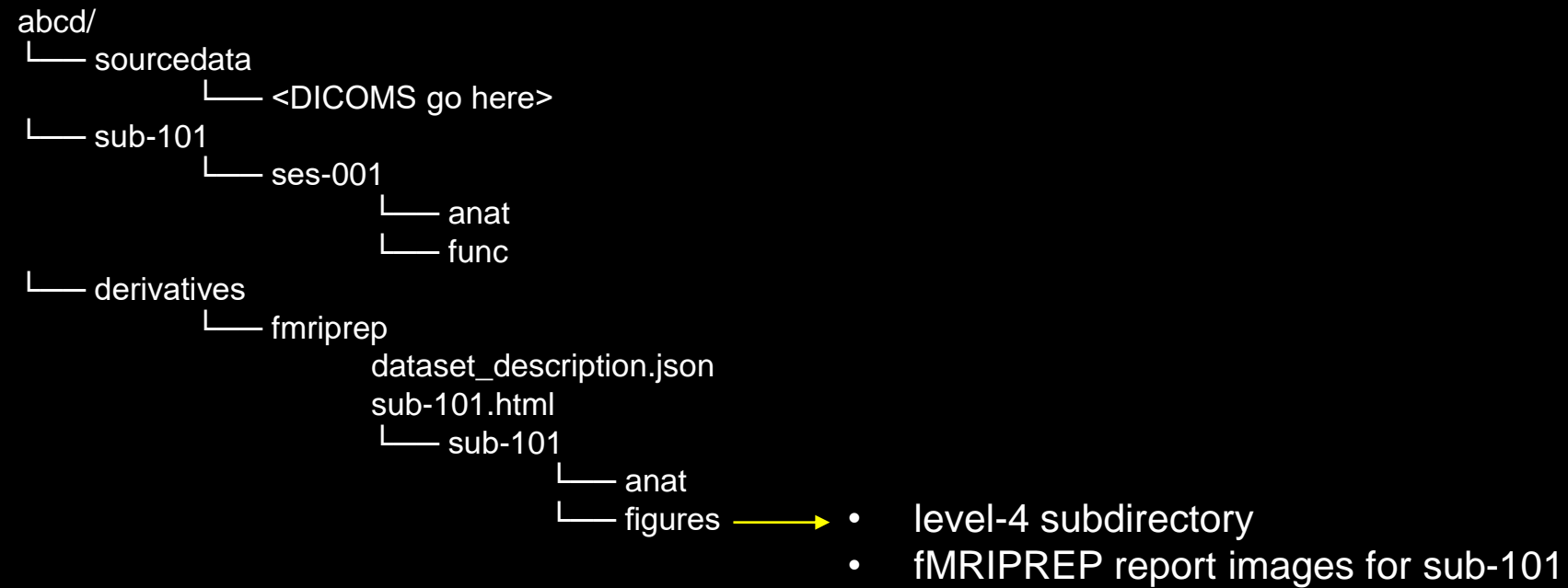
BIDS



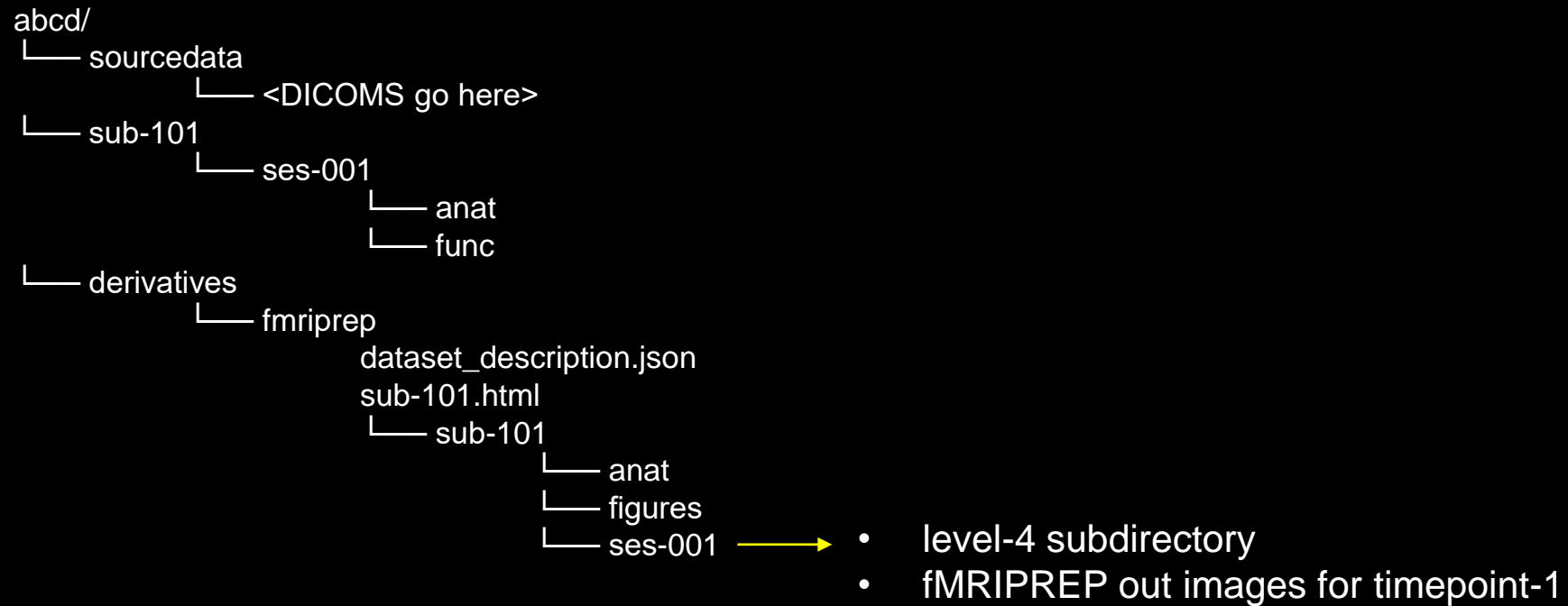
BIDS



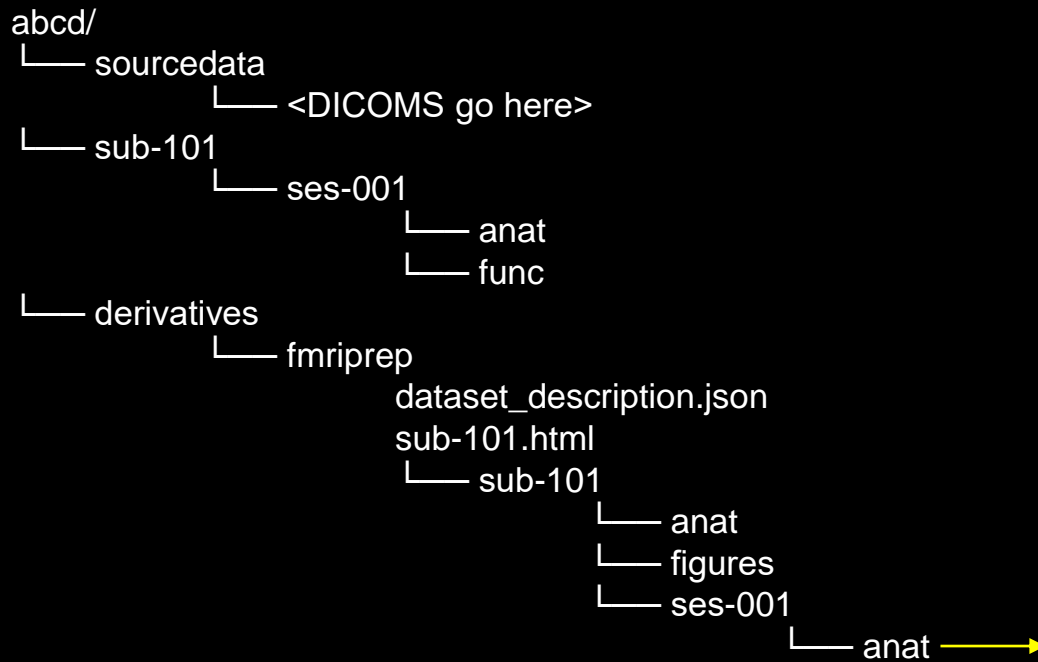
BIDS



BIDS

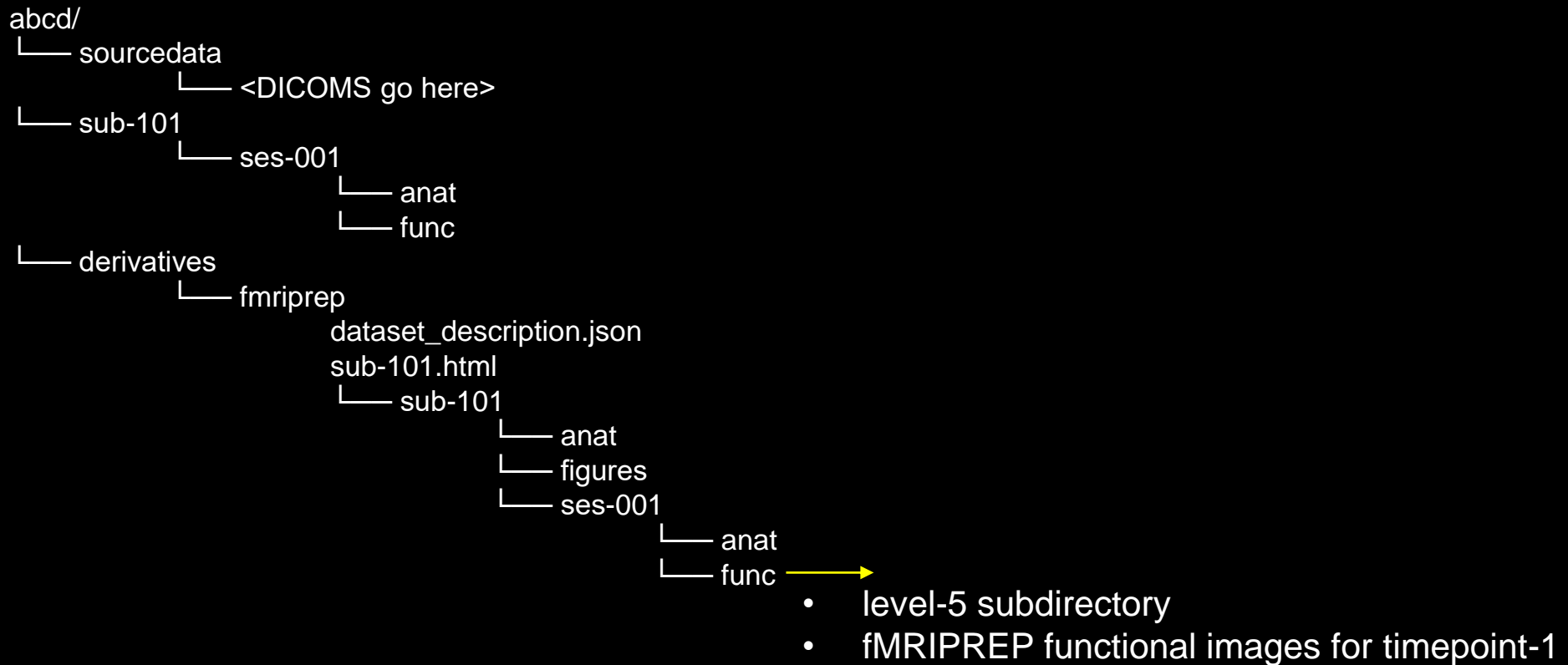


BIDS

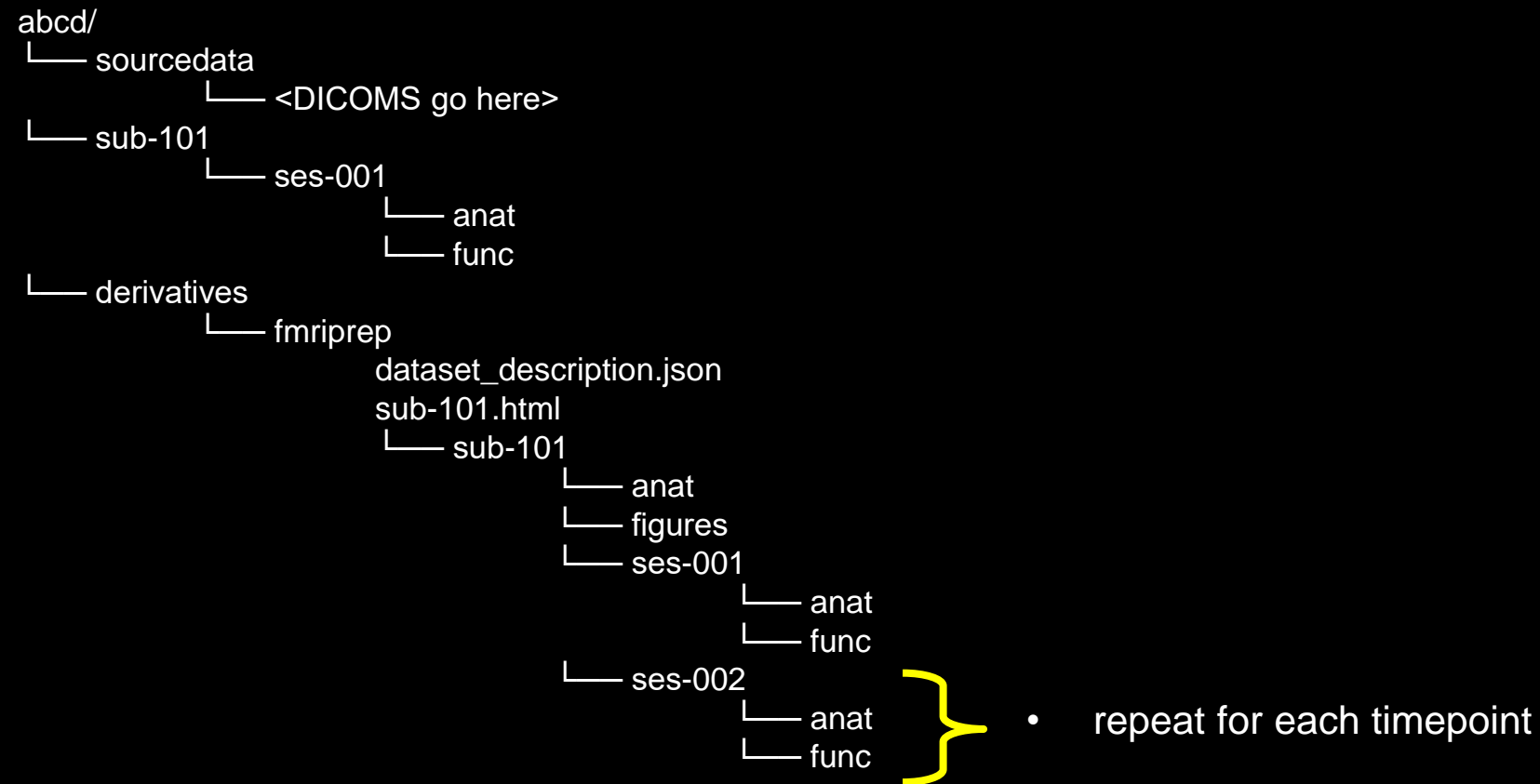


- level-5 subdirectory
- fMRIPREP anatomical images for timepoint-1

BIDS



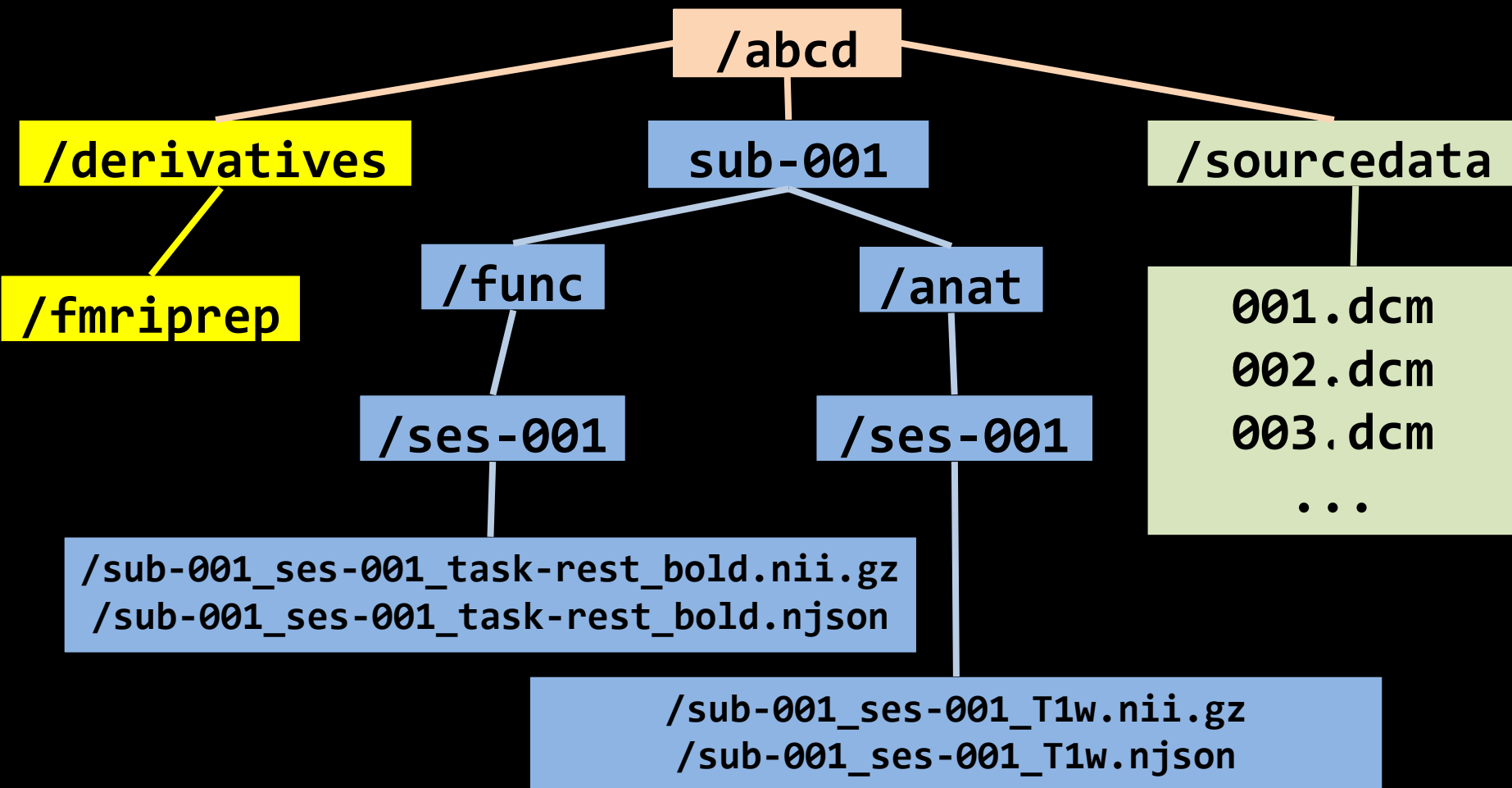
BIDS



Instruction #13

13. While that job process (should take about 2 hours), lets review an existing output:

```
[ ]$ cd ../derivatives  
[ ]$ firefox sub-tutorial.html
```

Additional BU RCS Web Resources

- Research Computing Support Pages
<http://www.bu.edu/tech/support/research/>
- Technical Summary of SCC Resources
<http://www.bu.edu/tech/support/research/computing-resources/tech-summary/>
- SCC Updates – Latest SCC News
<http://www.bu.edu/tech/support/research/whats-happening/updates/>
- Code Examples for Popular Software Packages
<http://scv.bu.edu/examples/>

1. Please open a web browser and:
2. Google “rcs tutorial evaluation”
3. Click 1st link
http://scv.bu.edu/survey/tutorial_evaluation.html
4. Fill out the survey!