

Brain Imaging Data Structure

BIDS and pieces – How to organize your data and why

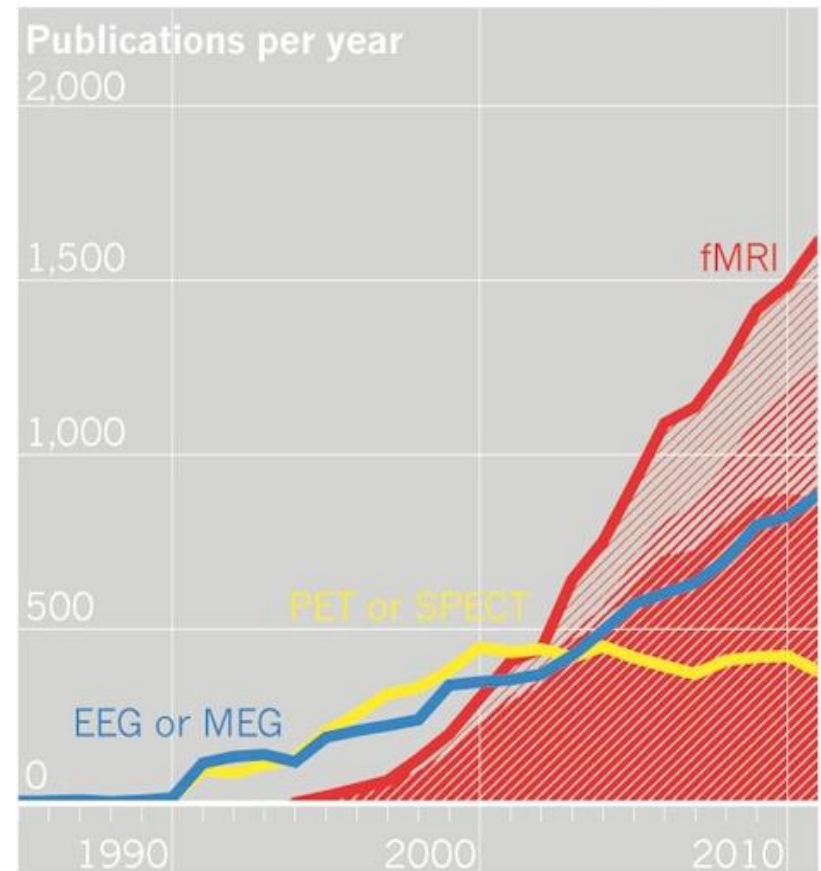
Tibor Auer

Research Fellow in MRI

Department of Psychology

Challenge

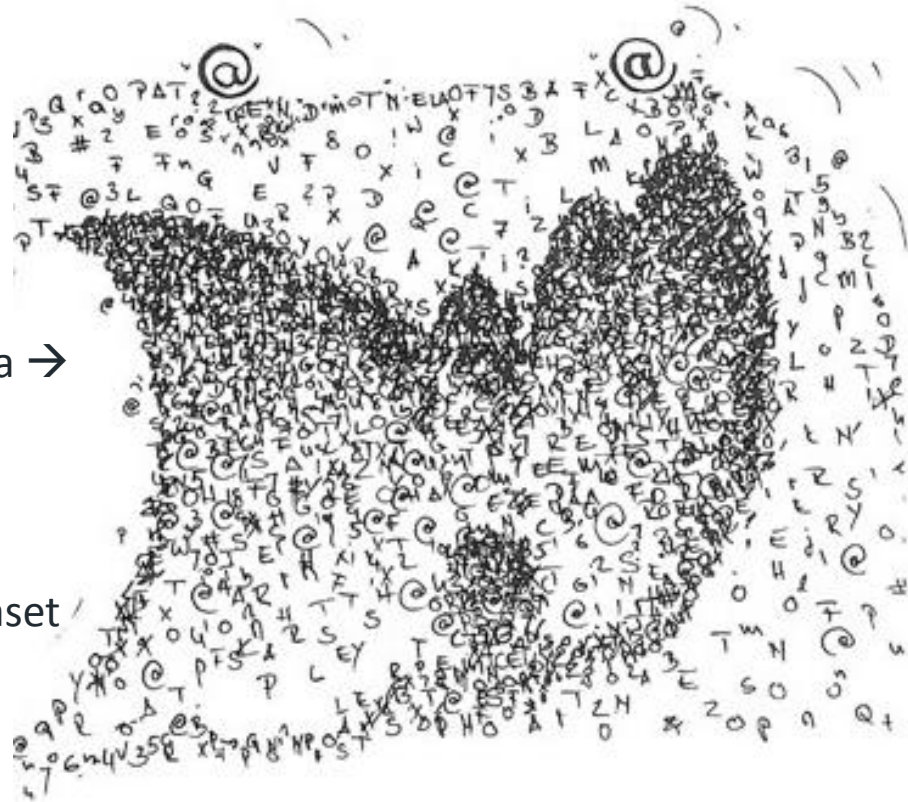
- **Lack of consensus**
 - Neuroimaging field generates an **increasing amount of data**
 - Neuroimaging experiments result in **complicated data**
 - Despite similarities in experimental designs and data types, each researcher tends to organize and describe their data in their own way



<http://www.nature.com/news/brain-imaging-fmri-2-0-1.10365>

Challenge

- **Getting lost in data**
 - Problems in data sharing
 - Within the same lab
 - Data repositories¹
↓
 - Fragmented efforts
 - „Unvisible” data
 - Rearranging data
 - Data processing is not aware of the data →
 - Unnecessary manual input
 - Rewriting scripts
 - Lack of automatic validation of the dataset
 - Accuracy
 - Completeness



BIDS

Introduction

- **BIDS**
 - Brain Imaging Data Structure (BIDS) is a new standard for organizing data of a human neuroimaging experiment.
 - <http://bids.neuroimaging.io>
- **Advantages for**
 - PI: More than one person working on the same data over time
 - User: Software aware of the data structure → automatic processing
 - Developer: Data structure, metadata can be expected
 - Database: Easier to include/share/exchange data
 - Some databases already accept BIDS
 - More grants/journals require data depositing/sharing
 - Validator tool

BIDS

Implementation

- **Principles**
 - Comprehensibility: metadata essential to capture most of experiments
 - Simplicity: no external software or complicated file formats
 - Flexibility: space to extend the standard

BIDS

Implementation

- **Comprehensibility**
 - Folder structure
 - Filename
- } some redundancy

sub-control01/

anat/

```
sub-control01_T1w.nii.gz
sub-control01_T1w.json
sub-control01_T2w.nii.gz
sub-control01_T2w.json
```

func/

```
sub-control01_task-nback_bold.nii.gz
sub-control01_task-nback_bold.json
sub-control01_task-nback_events.tsv
sub-control01_task-nback_cont-physio.tsv
sub-control01_task-nback_cont-physio.json
sub-control01_task-nback_sbref.nii.gz
```

dwi/

```
sub-control01_dwi.nii.gz
sub-control01_dwi.bval
sub-control01_dwi.bvec
```

fmap

```
sub-control01_phasediff.nii.gz
sub-control01_phasediff.json
sub-control01_magnitude1.nii.gz
sub-control01_scans.tsv
```

README

CHANGES

dataset_description.json

participants.tsv

BIDS

Implementation

- **Comprehensibility**

- Folder structure
 - Filename
 - JSON files for key–value pairs
- } some redundancy


```
{  
  
  "RepetitionTime": 3.0,  
  
  "EchoTime": 0.03,  
  
  "FlipAngle": 78,  
  
  "SliceTiming": [0.0, 0.2, 0.4, 0.6, 0.8, 1.0, 1.2, 1.4,  
                  1.6, 1.8, 2.0, 2.2, 2.4, 2.6, 2.8],  
  
  "InPlanePhaseEncodingDirection": "AP"  
  
  "TaskName": "nback"  
  
}
```

BIDS

Implementation

- **Simplicity**

- Use of compressed NIFTI files for imaging data. 

- Use of tab separated files for tabular data (demographics, events). 

```
participant_id age sex
sub-001         34  M
Sub-002         12  F
Sub-003         33  F
```

```
onset duration trial_type ResponseTime
1.2    0.6      go           1.435
5.6    0.6      stop          1.739
...
```

- Use of legacy text file formats for b vectors/values

BIDS

Implementation

- **Flexibility**
 - Handles multiple sessions and runs
 - Make certain folder hierarchy levels optional for simplicity.
 - Supports multiple types of anatomical scans
 - Supports fMRI: both task based and resting state.
 - Supports sparse fMRI (via slice timing)
 - Supports multiple fieldmap formats
 - Supports diffusions data (together with corresponding bvec, bval files)
 - Allows for arbitrary files not covered by the spec to be included.
 - Supports behavioural variables on any level (subjects, sessions and runs).
 - Supports contiguous acquisition covariates (breathing, cardiac etc.)

BIDS

Solutions

- **Community involved**
 - **Poldrack Lab at Stanford**
 - International Neuroinformatics Coordinating Facility (INCF), Neuroimaging Data Sharing Task Force (NIDASH-TF)
- **Validation tool:** <https://github.com/Squishymedia/bids-validator>
 - Browser-based
 - Via command line
- **Databases**
 - COINS, LORIS , OpenfMRI.org, SciTran , XNAT
- **Pipelines**
 - aa, C-PAC, Nipype
 - **BIDS Apps**

aa

Solutions

- **aa to process BIDS:**

% Add data

```
aap.directory_conventions.rawdatadir = '/imaging/ta02/Temp/BIDS/ds114';
```

```
aap = aas_processBIDS(aap);
```

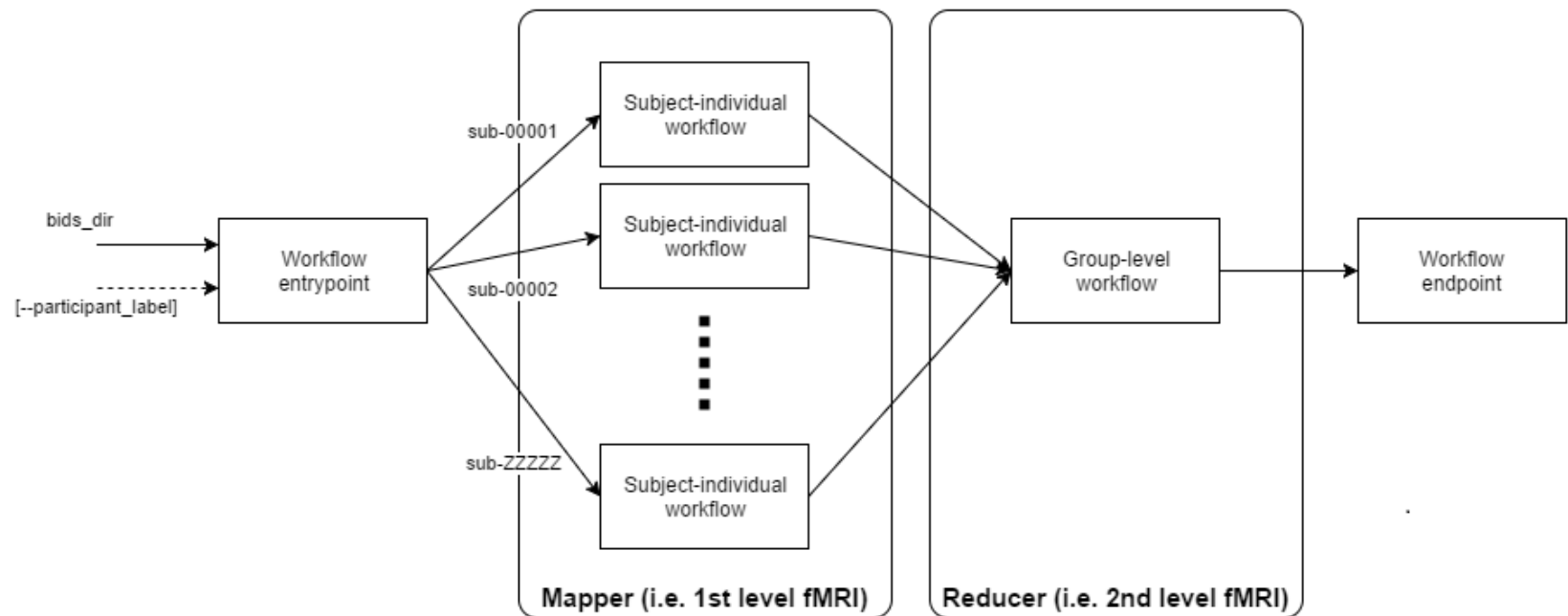


- For: functional, structural, diffusion
- Adds subjects
- Adds sessions
- Adds events

BIDS Apps

Solutions

- **BIDS Apps**
 - Portable neuroimaging pipelines that understand BIDS datasets
 - <http://bids-apps.neuroimaging.io>



<http://dx.doi.org/10.1101/079145>

BIDS Apps

Reproducibility – Concept

- **Portability**
 - Reproduce analysis in any working environment
 - **Versioning and archiving (provenance)**
 - Reproduce previous analyses
 - Transparency
 - **Fast adoption and automation**
 - Reproduce analysis efficiently
 - Education
 - **Creating and testing**
 - Develop analysis pipelines efficiently
-
- The diagram uses orange curly braces on the right side to group the concepts into two categories:
- Application:** This category includes the first two main concepts: **Portability** (with its sub-point 'Reproduce analysis in any working environment') and **Versioning and archiving (provenance)** (with sub-points 'Reproduce previous analyses' and 'Transparency').
 - Development:** This category includes the last two main concepts: **Fast adoption and automation** (with sub-points 'Reproduce analysis efficiently' and 'Education') and **Creating and testing** (with sub-point 'Develop analysis pipelines efficiently').

BIDS Apps

Reproducibility – Framework

- **Portability – Container**

- Docker (<http://docker.com>)
- Encapsulates all dependencies¹ in one convenient package
- Runs on all three major operating systems without setup and configuration



- **Versioning and archiving – Store**

- GitHub: <http://github.com/BIDS-Apps>
- Docker Hub: <https://hub.docker.com/u/bids>



- **Fast adoption and automation – Syntax**

- Same core set of obligatory command line arguments
 - bids_dataset
 - output
 - analysis_level: *participant [--participant_label 01] / group*

- **Creating and testing – Deployment**

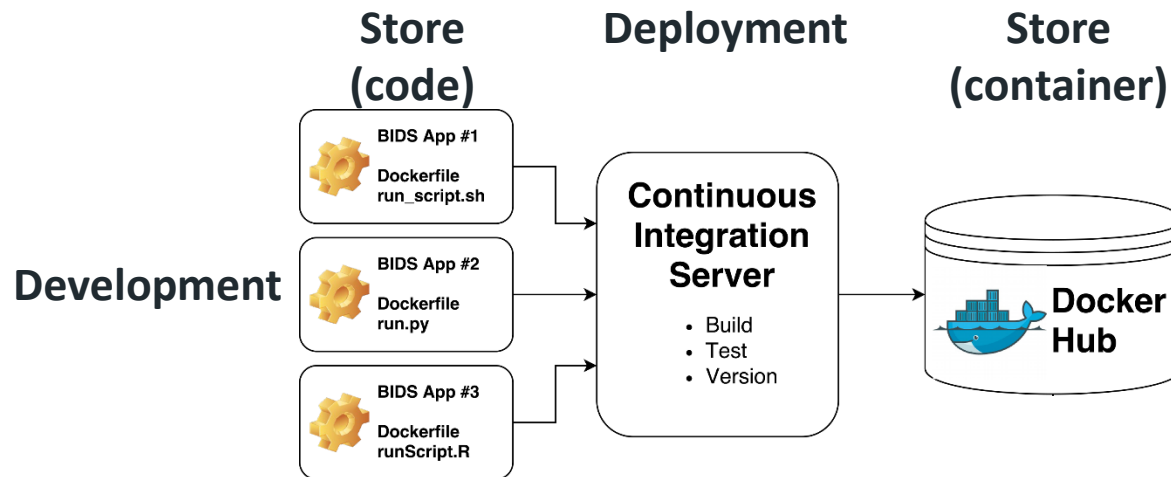
- Continuous Integration Server (<https://circleci.com/gh/BIDS-Apps>)



BIDS Apps

Reproducibility – Framework

- **BIDS Apps**
 - Portable neuroimaging pipelines that understand BIDS datasets
 - <http://bids-apps.neuroimaging.io>



<http://dx.doi.org/10.1101/079145>

- **Stanford Center for Reproducible Neuroscience**
 - Open platform for
 - data and methods sharing
 - reproducible analysis with high-performance
 - <http://reproducibility.stanford.edu>, <http://prod-openfmri.tacc.utexas.edu>
- **Data:** store, share, access
 - OpenfMRI → BIDS
- **Analysis:** (re)process data
 - BIDS Apps
 - Singularity HPC: parallelized across participants (+)
- **Quantifying the reproducibility of the results**
 - Across data
 - Across pipelines
 - Across versions



Reproducibility – Investigation

- **The OHBM Replication Award**
 - 2000 USD to the best published replication study of the past year
 - <http://reproducibility.stanford.edu/award>
- **Replication study**
 - Repetition of a published study
 - With minor changes assumed not to be important for the measured phenomena
 - Openness:
 - Obligatory: data, methods, results, deposited preprint (e.g. biorXiv)
 - Desirable: pre-registration, discussion with the original authors