Lifebrain

D3.3: MULTI-MODAL MRI-DATA PROCESSING STREAM

Project title:	Healthy minds from 0 to 100 years: Optimising the use of European brain imaging cohorts
Due date of deliverable:	31th December, 2017
Submission date of deliverable:	21th December, 2017
Leader for this deliverable:	University of Oslo

This project has received funding from the *European Union's Horizon2020 research and innovation programme* under grant agreement *No 732592.*

Contributors to deliverable:	Name	Organisation	Role / Title
Deliverable Leaders	Athanasia Monika Mowinckel	UiO	Researcher
	Inge Amlien	UiO	Senior engineer
Evaluation	Anders M. Fjell	UiO	WP6 leader
	Klaus Ebmeier	UOXF	PI
	William Baaré	RegionH	WP2 leader
	Sandra Düzel	MPIB	Researcher
	Rogier Kievit	MRC	Researcher
	David Bartrés-Faz	UB	PI
	Cristina Solé Padullés	UB	Researcher
	Lars Nyberg	UmU	PI
	Micael Andersson	UmU	Researcher
Final review and approval	Barbara B. Friedman	UiO	Administrative coordinator

Document History				
Release	Date	Reason for Change	Status (Draft/In-review/Submitted	Distribution
1.0.	13.12.2017	First draft for the General Assembly	Sent for commenting	OneDrive
2.0.	21.12.2017	Finalised based on the GenA comments	Final deliverable	Participant portal

Dissemination level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

EXECUTIVE SUMMARY

For a project like Lifebrain to succeed, running analyses across sites and having data collected in a multitude of ways, the development, sharing and deployment of specialized analytical tools requires transparent and dynamic systems. This is not easy, and has been difficult for many scientific consortia to manage.

Several recent technological developments are timely for Lifebrain, because there are new tools, making management and distribution of processing streams easier. These developments are both regarding storage structure of complex data, and software for distribution, running, and tracking changes to analytical pipelines.

The following is a proposal for several services or software that together create a transparent and collaborative system for developing, tracking and sharing (neuroimaging) processing streams.

The proposed stream is based on a BIDS – Brain Imaging Data Structure. Containers would be used for distributing processing pipelines that can run without virtual environments. The structure will also include all necessary software and libraries to run the analysis. This will create unified analyses across sites until data may be shared.

TABLE OF CONTENTS

LIST OF ACRONYMS/ ABBREVIATIONS

BIDS	Brain Imaging Data Structure
EB	Executive Board
GA	Grant Agreement
GenA	General Assembly
LB	Lifebrain
Lifebrain	Healthy minds from 0-100 years: Optimising the use of European brain imaging cohorts
M	Month
MPIB	Max Planck Institute for Human Development
MRI	Magnetic Resonance Imaging
PI	Principal Investigator
REGIONH	Region Hovedstad
TSD	Services for Sensitive Data
UB	University of Barcelona
UCAM	University of Cambridge
UiO	University of Oslo
UOXF	University of Oxford
UmU	Umeå University
XNAT	Extensible Neuroimaging Archive Toolkit
WP	Work Package

1. INTRODUCTION

1.1. DESCRIPTION OF DELIVERABLE

Task 3.3 Development of multi-modal MRI-data processing stream. Lead: UiO; Participants: UOXF, REGIONH, UmU, MPIB (M1-M12)

A standardized analysis stream will be developed that all data can be run through, with special focus put on neuroimaging. We will develop a common framework for pre-processing and statistical analyses of multi-modal brain scans, including morphometry, DTI and fMRI. This will be based on combining and integrating features from state-of-the-art FSL (developed at fMRIB, UOXF) and Freesurfer (developed at Harvard) software with custom-made procedures for this project. The data processing stream will be provided with the central hub of the data transfer, storage and processing system (T3.1) to allow for standardized, integrated processing of data across all cohorts. Output of this processing stream can be fed into further statistical modelling.

1.2. OBJECTIVE

D3.3: Multi-modal MRI-data processing stream which allows for a state-of-the art central integration of data across sites [12]

A standardized analysis stream will be developed that all data can be run through, with special focus put on neuroimaging.

1.3. COLLABORATION AMONG PARTNERS

A proposal on a multi-modal MRI data-processing stream has been presented to the consortium by UiO at the Lifebrain project meeting in Barcelona, 16th November, 2017. FreeSurfer 6 was accepted as a software solution by the GenA for processing MRI images. A decision has been also made by the GenA on the necessity of acquiring a GitHub.com repository service account.

A document describing the stream was then drafted in December, 2017 and shared with the GenA. The document has been revised based on the feedbacks from the GenA and finalised.

2. BACKGROUND

For a project like Lifebrain to succeed, running analyses across sites and having data collected in a multitude of ways, the development, sharing and deployment of specialized analytical tools requires transparent and dynamic systems. This is not easy, and has been difficult for many scientific consortia to manage.

Several recent technological developments are timely for Lifebrain, because there are new tools, making management and distribution of processing streams easier. These developments are regarding both storage structure of complex data, and software for distribution, running, and tracking changes to analytical pipelines.

The following is a proposal for several services or software that together create a transparent and collaborative system for developing, tracking and sharing (neuroimaging) processing streams.

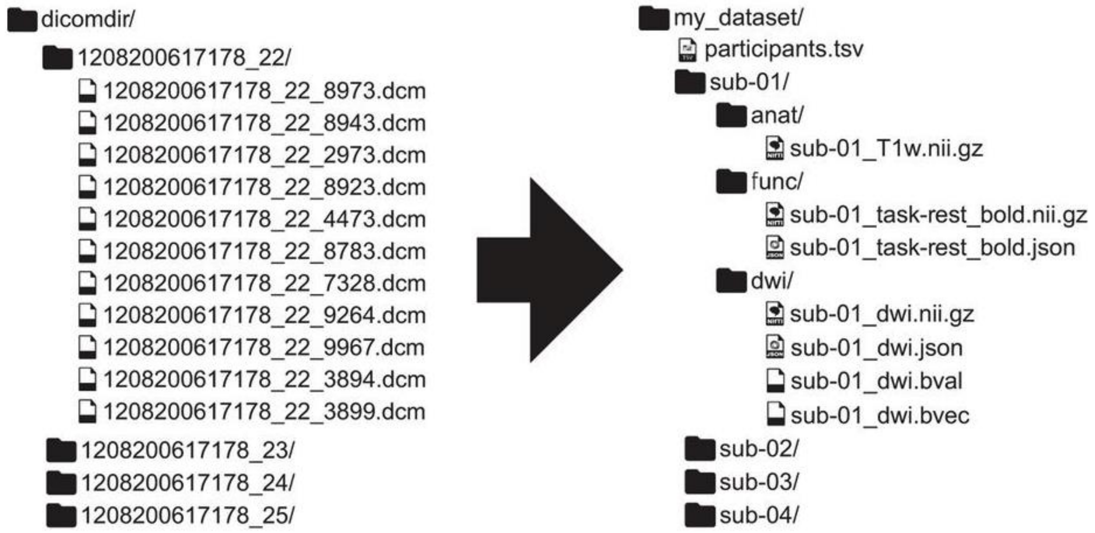
3. BIDS – BRAIN IMAGING DATA STRUCTURE

Neuroimaging data come in a variety of types; longitudinal and cross-sectional, with and without experimental manipulations, with different recorded modalities etc. There is no consensus on how to structure the storage of such data, and neuroimaging datasets are handled in an extremely heterogeneous fashion. This makes it hard for researchers to share and navigate each other’s datasets. However, the recently developed Brain Imaging Data structure (BIDS) promises a simple and easy-to-adopt way of organizing neuroimaging and behavioural data (Gorgolewski et al. 2016). BIDS is purely an organizational structure for data, not any type of software, and as such there are no limits imposed by the structure regarding which software that may be used with the data.

3.1. BENEFITS OF USING BIDS

- It provides a simple and efficient structure to store and access data
- It already has thorough user-documentation
- There is a growing number of data analysis software packages that can "understand" data organized according to BIDS
- Existing database platforms such as OpenfMRI.org, LORIS, COINS, XNAT, and SciTran accept and export datasets organized according to BIDS
- Ease of sharing data with publishers, who require this, due to its simple structure
- Validation tools (also available online) allow to check and report on dataset integrity
- Conversion tools allow e.g. to convert OpenfMRI style organized data to BIDS

The BIDS is heavily inspired by the format used internally by OpenfMRI.org. The specification intentionally uses simple file formats and folder structures to reflect current lab practices. This makes BIDS highly accessible and applicable to a wide range of scientists coming from different backgrounds.


dicomdir/
1208200617178_22/
1208200617178_22_8973.dcm
1208200617178_22_8943.dcm
1208200617178_22_2973.dcm
1208200617178_22_8923.dcm
1208200617178_22_4473.dcm
1208200617178_22_8783.dcm
1208200617178_22_7328.dcm
1208200617178_22_9264.dcm
1208200617178_22_9967.dcm
1208200617178_22_3894.dcm
1208200617178_22_3899.dcm
1208200617178_23/
1208200617178_24/
1208200617178_25/
my_dataset/
participants.tsv
sub-01/
anat/
sub-01_T1w.nii.gz
func/
sub-01_task-rest_bold.nii.gz
sub-01_task-rest_bold.json
dwi/
sub-01_dwi.nii.gz
sub-01_dwi.json
sub-01_dwi.bval
sub-01_dwi.bvec
sub-02/
sub-03/
sub-04/

BIDS is a format for standardizing and describing outputs of neuroimaging experiments (left) in a way that is intuitive to understand and easy to use with existing analysis tools (right).

image source

3.2. STRUCTURE

According to BIDS, data is stored in a two or three tiered, nested folder structure, with each participant having all their data within one folder, which further may be divided into session folders for each subsequent visit. There is much flexibility as to how to define a “session”, as this label may be used for separating longitudinal data, repeated measures, experimental conditions etc., and inherently depends on the data at hand.

Within the session folder, folders for anatomical (anat), functional (func), diffusion weighted (dwi) and behavioural (behav) data exists to divide different modalities into different folders. Additional information paired to scans may be placed in JavaScript Object Notation (JSON) files, to provide extra information. Tab-separated (.tsv) files may also be placed in different folders within the hierarchy to provide information regarding data to be expected within the tiers. For instance, at the top level a “Participant.tsv” should exist listing all participants, within a participant’s folder, another tsv may be placed to provide overview of the sessions etc.

3.3. SOFTWARE CURRENTLY SUPPORTING BIDS

A growing set of portable containerized data (see 4.1. Containers and images) processing pipelines are compatible with datasets structured according to BIDS. Given BIDS organized data, pre-compiled analytical pipelines are easy to implement, without major customization of processing workflows to account for different naming schemes and parameters across imaging sites.

BIDS apps can be run both on local systems and on High Performance Clusters (HPC), like the UiO Colossus cluster. For more information on analytical set up through container images, see the section 4.1. Containers and images.

BIDS apps include apps for:

- Human Connectome Project analysis pipelines (https://github.com/BIDS-Apps/HCPPipelines),
- Extraction of resting state metrics (https://github.com/BIDS-Apps/rs_signal_extract),
- Running Freesurfer's recon-all (https://github.com/BIDS-Apps/freesurfer),
- State-of-the-art quality assessment analyses (https://github.com/BIDS-Apps/QAP/ & https://github.com/BIDS-Apps/mriqc).

There is furthermore a tool for integration of BIDS and XNAT, which is particularly convenient for sites already using XNAT, and for the further use of XNAT in Lifebrain.

See Appendix 1 for a list of relevant BIDS Apps, and an updated list of currently released BIDS apps at: http://bids-apps.neuroimaging.io/apps/

By adapting the BIDS structure, all these apps are readily run on Lifebrain data, providing excellent utilization of the newest developed pipelines without installing a plethora of software or specialized scripts to make the apps work with our own data organisational structure.

The processing, analytical and integration pipelines developed by the Lifebrain Consortium can be shared with and distributed to a wider neuroimaging community. The efforts laid down in developing these tools may thus represent methodological contributions that can be refined and built upon by the general scientific community. Information on what needs to be taken into account when developing BIDS apps is available online.

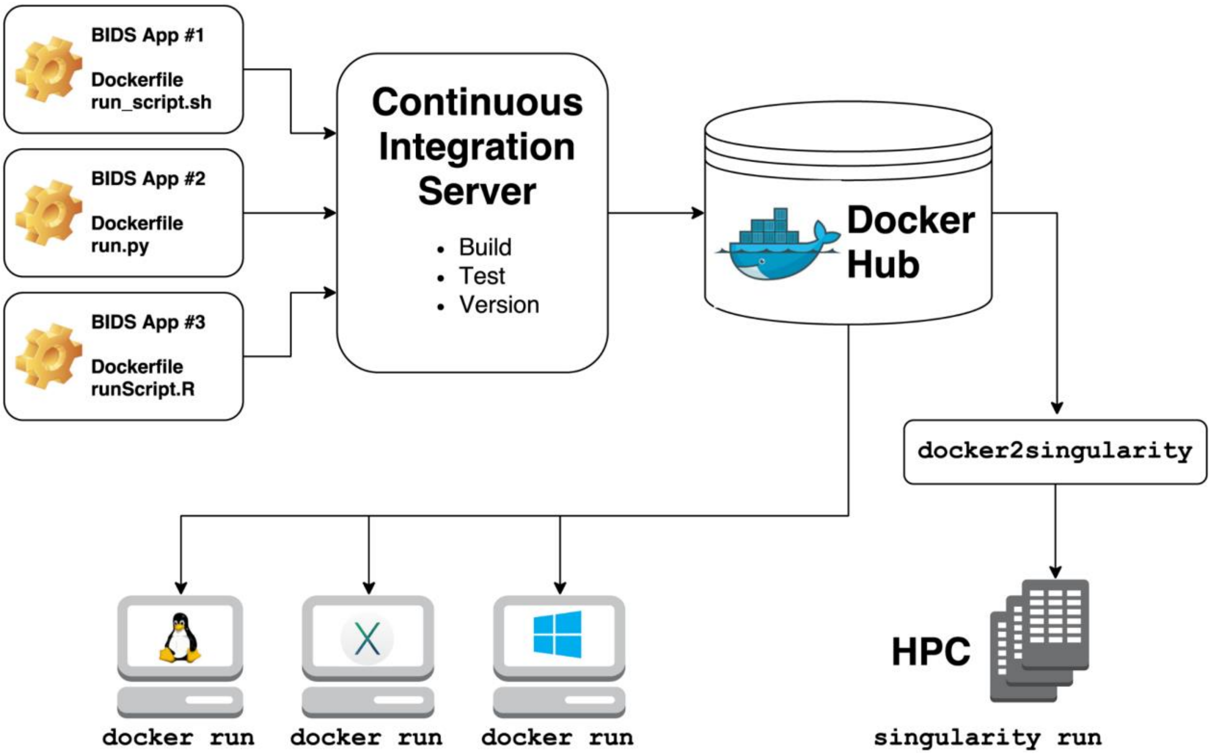


FIGURE 1 - OVERVIEW OF THE CREATION AND USE OF BIDS APPS.

The source code of each app is stored in separate github repository. each repository is connected to a continuous integration server responsible for building testing and deploying the corresponding app. for every new release of an app, a new container image is deposited in docker hub. users can directly download and run the bids apps container images either directly using docker on any windows, mac, and linux machine or convert them to singularity and run them on an hpc. image source

4. DEVELOPMENT AND DISTRIBUTION OF MULTI-MODAL MRI PROCESSING STREAMS AND PIPELINES

Whereas all Lifebrain partners have agreed that they are allowed and able to transfer raw imaging data to the centralized storage solution at UiO (TSD), the development of state-of-the-art analytical pipelines is focused on creating pipelines that can run locally, as well as centrally. This makes it possible to distribute analytical pipelines prior to actual data sharing, which might increase data production.

A major challenge we face with the development of a standardized analytical stream is to ensure that data are processed identically across sites. Whereas unlikely to have large effects, different computing environments have been shown to influence analytical results (for example as shown in Gronenschild et al. 2012). The proposed solution to this challenge is to execute the imaging analytical pipelines utilizing container technology, such as Docker. The container approach ensures that all data are processed identical across sites, regardless of underlying computing architecture.

Using this approach, we will cooperatively develop and integrate container images that perform all processing steps from raw data to tabular statistics files, and output standardized QA metrics. Containers can be developed with analytical streams for each modality separately, with multimodal integration at the end-point. This approach requires that most sites are able to run container or singularity images, and that input data are organized in accordance with pre-defined schemes (BIDS).

4.1. CONTAINERS AND IMAGES

An **image** is a lightweight, stand-alone, executable package that includes everything needed to run a piece of software, including the code, a runtime, libraries, environment variables, and configuration files.

A **container** is a runtime instance of an image — what the image becomes in memory when actually executed. It runs completely isolated from the host environment by default, only accessing host files and ports if configured to do so.

Typically, each container provides a single service (often called a "micro-service") such as a web server or a database, although containers can be used for arbitrary workloads. Containers run apps natively on the host machine's kernel. Containers can get native access, each one running in a discrete process, taking no more memory than any other executable. This makes containers faster, requiring less resources, than for instance running virtual machines.

A container example is the data-explorer (ShinyApp) that has been developed for Lifebrain. This data explorer is run by a container image of R, and accessed via an internet browser. This explorer is to be used by partners within the central storage solution (TSD), directly linked to the Lifebrain database (tabular data). The prototype of this ShinyApp has been launched online, with a mock dataset already loaded.

4.2. ONLINE REPOSITORIES AND VERSION CONTROL

Another challenge in working across sites, is the distribution and tracking of tool development. Whereas sending scripts and processing pipelines via e-mail is possible, it provides little control or traceability, and requires the developer to send out mass e-mails each time a new feature or bug has been fixed.

The distribution and development of processing streams across sites is simplified through the use of online cloud/repository services. The benefit of such services is that scripts can be made easily available to all partners on an online server, so tools may be easily downloaded and implemented. Furthermore, with version control integration, it also makes the development of tools transparent, such that all changes to the tools as they are implemented are traced, and there are simple ways to restore previous versions if a new release has a problem. It also allows several researchers to collaborate on tool development, while tracing who is making which alterations.

An academic GitHub.com account has been acquired, where Lifebrain has unlimited number of private repositories and users at no cost. GitHub is likely the most known and widely used version control repository service. It is easy to use and several Lifebrain members have already started collaborating on this platform.

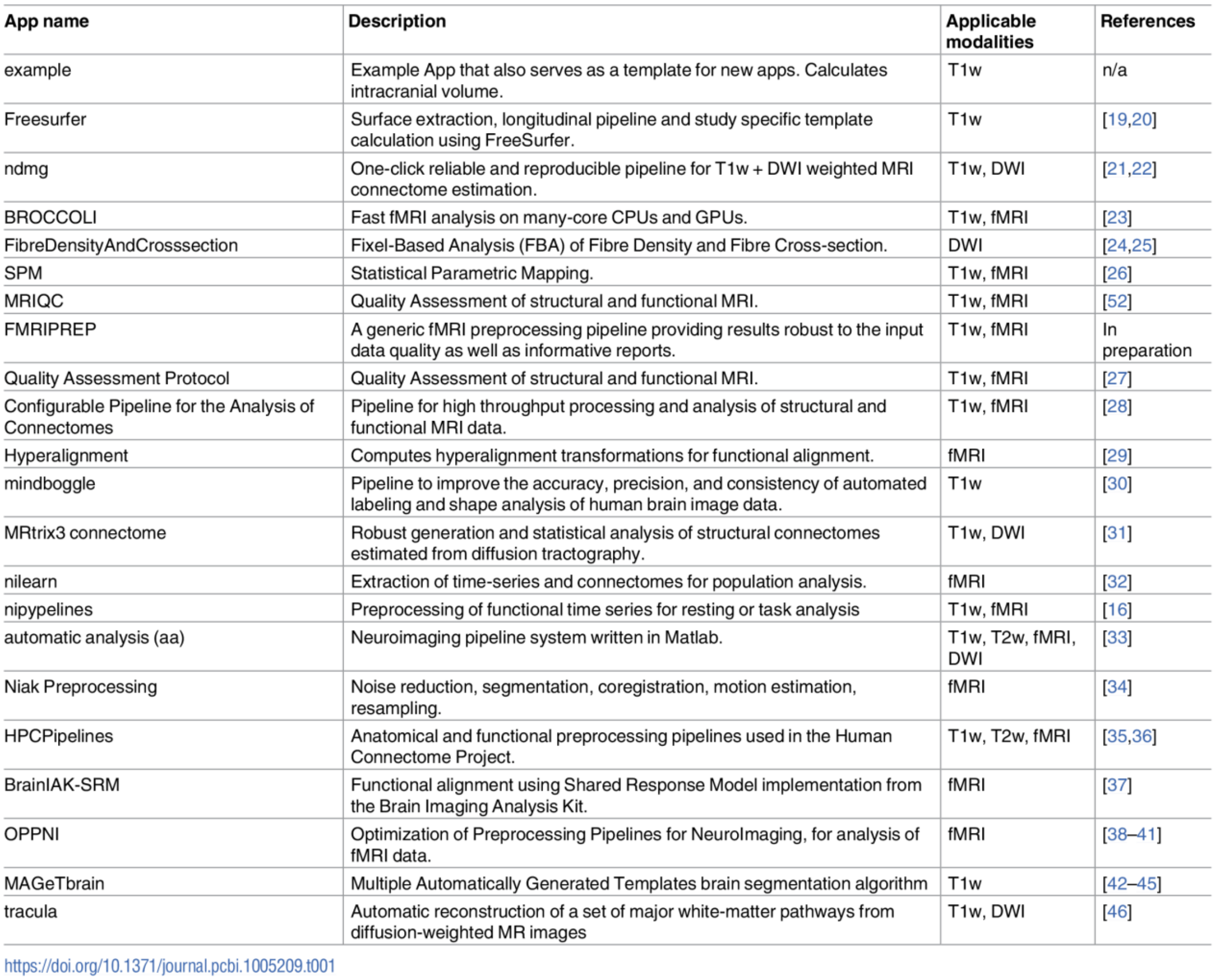
4.3. NEUROIMAGING SOFTWARE

The system described in this document focuses on creating an environment where development and distribution of analysis streams can occur in transparent and efficient manner, rather than describing a specific approach for a multi-modal processing stream. There are multitudes of different MRI processing tools available for use, each with its own merit, and focus regarding modality. Given the heterogeneity of MRI-data and analysis expertise in Lifebrain, it is difficult to *a priori* decide which tools are the most suitable. The software highlighted in the following section functions more as references to software that there is expertise in using within Lifebrain, and thus would be the first tools to be tested.

The Lifebrain GenA decided (November 16th, 2017) that FreeSurfer would be used for the processing and segmenting of structural T1 weighted images. Most partner sites already have their data processed by some version of FreeSurfer, and an initial focus will be to run full processing (recon-all) of all MRI data with FreeSurfer v.6.0 to increase homogeneity.

Other than FreeSurfer, FSL is also likely to be utilized for analyses of diffusion weighted data and functional data. DWI analysis usually requires images to be acquired with more homogenous acquisition parameters than what Lifebrain has at hand, and it is thus also possible that this particular modality would best be analysed in a meta-analytical approach.

While specialized analysis pipelines are under development, the adoption of the BIDS structure on imaging data allows the consortium to utilize the many tools already developed to run easily on data organized in this particular fashion, and also accelerate the internal tool development process. The different BIDS apps and tools (Appendix 1 - BIDS Applications) cover many modalities and most also output tabular summary data. While awaiting more specialized tools, BIDS apps provide various types of metrics (quality controls, cortical volume and thickness etc.) that will be used to provide initial data to the partners. There are already apps available for analytical approaches the consortium has interest in running, like FreeSurfer's recon-all for T1 weighted data and tracula for diffusion weighted data.



App name	Description	Applicable modalities	References
example	Example App that also serves as a template for new apps. Calculates intracranial volume.	T1w	n/a
Freesurfer	Surface extraction, longitudinal pipeline and study specific template calculation using FreeSurfer.	T1w	[19,20]
ndmg	One-click reliable and reproducible pipeline for T1w + DWI weighted MRI connectome estimation.	T1w, DWI	[21,22]
BROCCOLI	Fast fMRI analysis on many-core CPUs and GPUs.	T1w, fMRI	[23]
FibreDensityAndCrosssection	Fixel-Based Analysis (FBA) of Fibre Density and Fibre Cross-section.	DWI	[24,25]
SPM	Statistical Parametric Mapping.	T1w, fMRI	[26]
MRIQC	Quality Assessment of structural and functional MRI.	T1w, fMRI	[52]
FMRIPREP	A generic fMRI preprocessing pipeline providing results robust to the input data quality as well as informative reports.	T1w, fMRI	In preparation
Quality Assessment Protocol	Quality Assessment of structural and functional MRI.	T1w, fMRI	[27]
Configurable Pipeline for the Analysis of Connectomes	Pipeline for high throughput processing and analysis of structural and functional MRI data.	T1w, fMRI	[28]
Hyperalignment	Computes hyperalignment transformations for functional alignment.	fMRI	[29]
mindboggle	Pipeline to improve the accuracy, precision, and consistency of automated labeling and shape analysis of human brain image data.	T1w	[30]
MRtrix3 connectome	Robust generation and statistical analysis of structural connectomes estimated from diffusion tractography.	T1w, DWI	[31]
nilearn	Extraction of time-series and connectomes for population analysis.	fMRI	[32]
nipypelines	Preprocessing of functional time series for resting or task analysis	T1w, fMRI	[16]
automatic analysis (aa)	Neuroimaging pipeline system written in Matlab.	T1w, T2w, fMRI, DWI	[33]
Niak Preprocessing	Noise reduction, segmentation, coregistration, motion estimation, resampling.	fMRI	[34]
HPCPipelines	Anatomical and functional preprocessing pipelines used in the Human Connectome Project.	T1w, T2w, fMRI	[35,36]
BrainIAK-SRM	Functional alignment using Shared Response Model implementation from the Brain Imaging Analysis Kit.	fMRI	[37]
OPPNI	Optimization of Preprocessing Pipelines for NeuroImaging, for analysis of fMRI data.	fMRI	[38–41]
MAGeTbrain	Multiple Automatically Generated Templates brain segmentation algorithm	T1w	[42–45]
tracula	Automatic reconstruction of a set of major white-matter pathways from diffusion-weighted MR images	T1w, DWI	[46]

https://doi.org/10.1371/journal.pcbi.1005209.t001

FIGURE 2 - LIST OF CURRENTLY AVAILABLE BIDS APPS. image source

5. IMPLEMENTATION IN LIFEBRAIN

The proposals above heavily rely on each other to function seamlessly. Developing analytical pipelines and distributing these will be particularly hard if data are not stored in a unified manner across sites, and sharing data will not reach full potential if we are unable to process all data as equal as possible.

We propose that each site, prior to transfer to the central storage and processing location, restructure their data to comply with the BIDS specification. The restructuring across all sites not only makes transfer and unification easier, but it also ensures all data that is transferred already has the desired structure for compiled analytical pipelines to be run in a timely and efficient manner. This distributes the work across sites, making the workload more manageable, in addition to avoiding complications due to underlying data being coded in languages unknown to the data manager. The exact format of the structure is based on the BIDS standard, but with some flexibility to account for challenges posed by the multi-centered nature of the project. Joint decisions regarding the final structure will be made before actual data transfer. See Appendix 2 for example data structure.

The BIDS already provides tools and apps for validating the conversion of pre-existing data, which should be straight forward to implement at each site prior to transfer. This means that neuroimaging data may be restructured and processed in a timely manner, and also reduces the number of potential failure modes considerably.

Further, various BIDS apps will be run on the data made available, and will thus provide homogenous tabular data from different MRI modalities in an efficient manner. The development of more specialized processing pipelines will run in parallel to this work.

6. REFERENCES

GitHub
https://github.com

Gorgolewski et al. (2017): BIDS apps: Improving ease of use, accessibility, and reproducibility of neuroimaging data analysis methods
http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005209

Gorgolewski et al. (2016): Brain Imaging Data Structure
http://bids.neuroimaging.io

Gronenschild et. al. 2012: The Effects of FreeSurfer Version, Workstation Type, and Macintosh Operating System Version on Anatomical Volume and Cortical Thickness Measurements
https://doi.org/10.1371/journal.pone.0038234

APPENDIX

1. BIDS APPLICATIONS

CONVERTERS

AFNI BIDS-tools

BIDS2ISATab

BIDSto3col

BIDS2NDA

Bidskit

dac2bids

Dcm2Bids

DCM2NIIx

DICM2NII

heudiconv

OpenfMRI2BID

QUALITY ASSURANCE / QUALITY CONTROL

MRIQC

QAP

PROCESSING TOOLS

FMRIPREP	Preprocessing workflow
Automatic Analysis	fMRI processing toolbox
C-PAC	Configurable Pipeline for the Analysing Connectomes
Human Connectome Pipeline	HCP Pipelines
FreeSurfer	Running recon-all
Tracula	Running TRACULA
SPM	SPM container

2. EXAMPLE DATA STRUCTURE

```
sub-0000001/
        ses-01/
                anat/
                        sub-0000001_ses-01_acq-UCAM01_mod-MPRAGE_run-01_T1w.nii.gz
                        sub-0000001_ses-01_acq-UCAM01_mod-MPRAGE_run-01_T1w.json
                        sub-0000001_ses-01_acq-UCAM01_mod-MPRAGE_run-02_T1w.nii.gz
                        sub-0000001_ses-01_acq-UCAM01_mod-MPRAGE_run-02_T1w.json
                        sub-0000001_ses-01_acq-UCAM01_mod-SPACE_run-01_T2w.nii.gz
                        sub-0000001_ses-01_acq-UCAM01_mod-SPACE_run-01_T2w.json

        ses-02/
                anat/
                        sub-0000001_ses-02_acq-UCAM03_mod-mprage_run-01_T1w.nii.gz
                        sub-0000001_ses-02_acq-UCAM03_mod-mprage_run-01_T1w.json
                        sub-0000001_ses-02_acq-UCAM03_mod-mprage_run-02_T1w.nii.gz
                        sub-0000001_ses-02_acq-UCAM03_mod-mprage_run-02_T1w.json
```