3-6 February 2020 - Athens, Greece

# Human Brain Project SUMMIT & OPEN DAY

Co-funded by
the European Union

Human Brain Project

# Developing pipelines for multi- scale/species/method analysis

## adaptable, reusable, UseCase-driven

05.02.2020 HBP Summit, *Athens*

Robin Gutzen, *Institute of Neuroscience and Medicine (INM-6), FZJ*

JÜLICH Forschungszentrum

Co-funded by the European Union

# Where we are - Where we want to be

Human Brain Project

## Towards large research collaborations and long-lasting projects

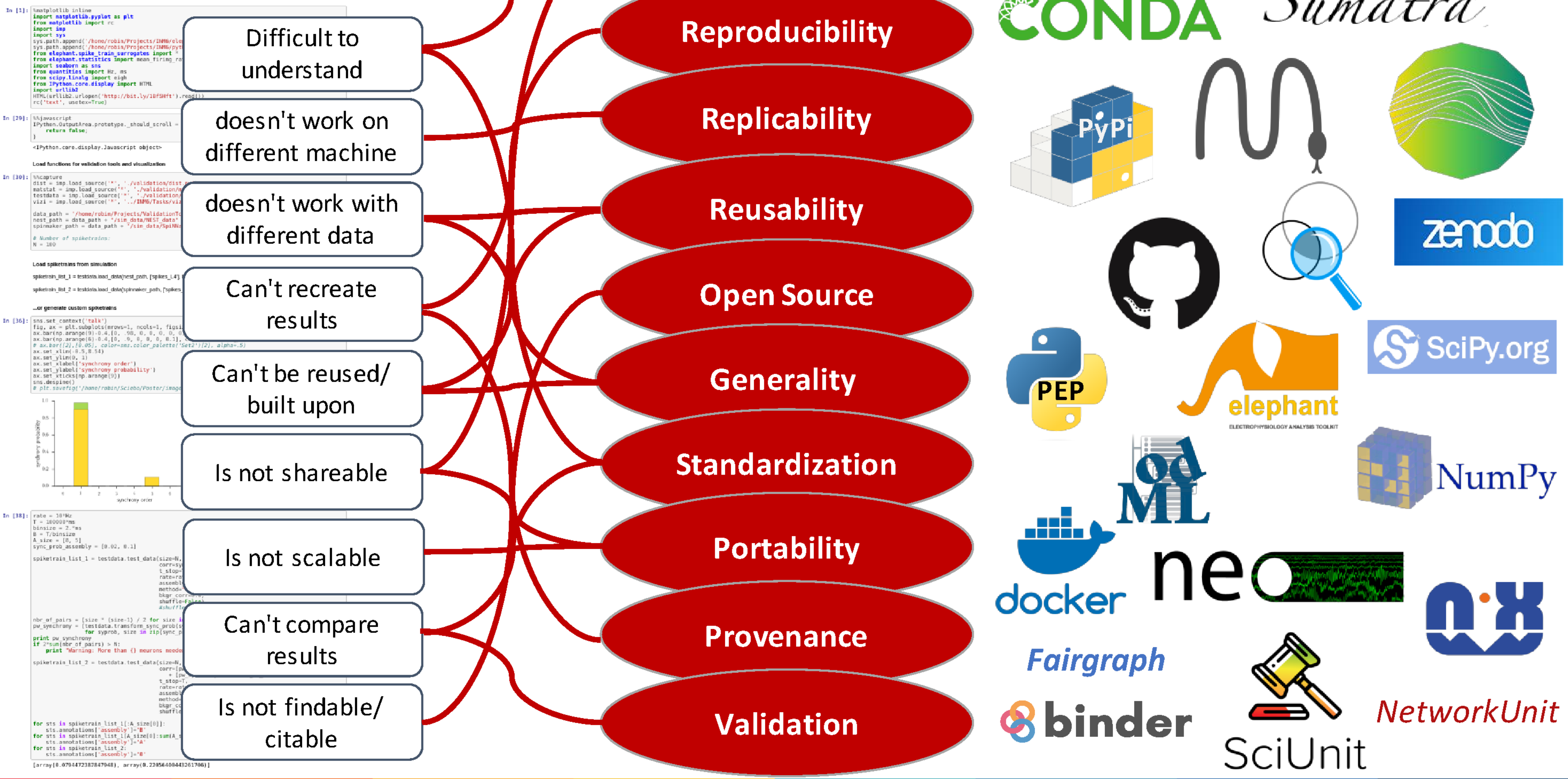
- Data sharing becomes easier and more common
- Open Access becomes more prominent
- "Reproducibility Crisis"

- Easy publishing of analysis workflows
- Easy replication of published results
- Easy reuse of published data/results/analysis

## How can we adapt our computational research practices?

- There is no one-size-fit-all solution
- A lot of little, incremental improvements
- Many solutions and tools already exist
- There is progress in bringing tools together
- There is progress in co-developing new features and bridges

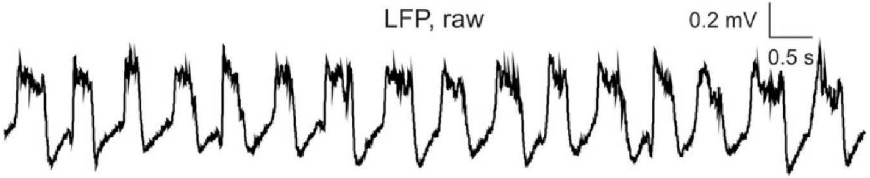
Human Brain Project



# Slow Wave Analysis (SP3 UseCase002)

## Slow Waves

- periodic transition between Up and Down state (0-5Hz)
- in sleep and anasthesia
- across measurements and species
- relevance for memory and consciousness research




*Szabó et. al. 2017*

## Analysis Pipeline

- identifying a generalized structure of analysis steps

  -> sequential *Stages*, and modular *Blocks*
- implemention with a workflow management sytem
- precisely defining the interfaces and (meta-)data requirements
- integrating existing analysis approaches & algorithms

Human Brain Project

# Flowchart

Pipeline | Stage | Block

Calcium Imaging

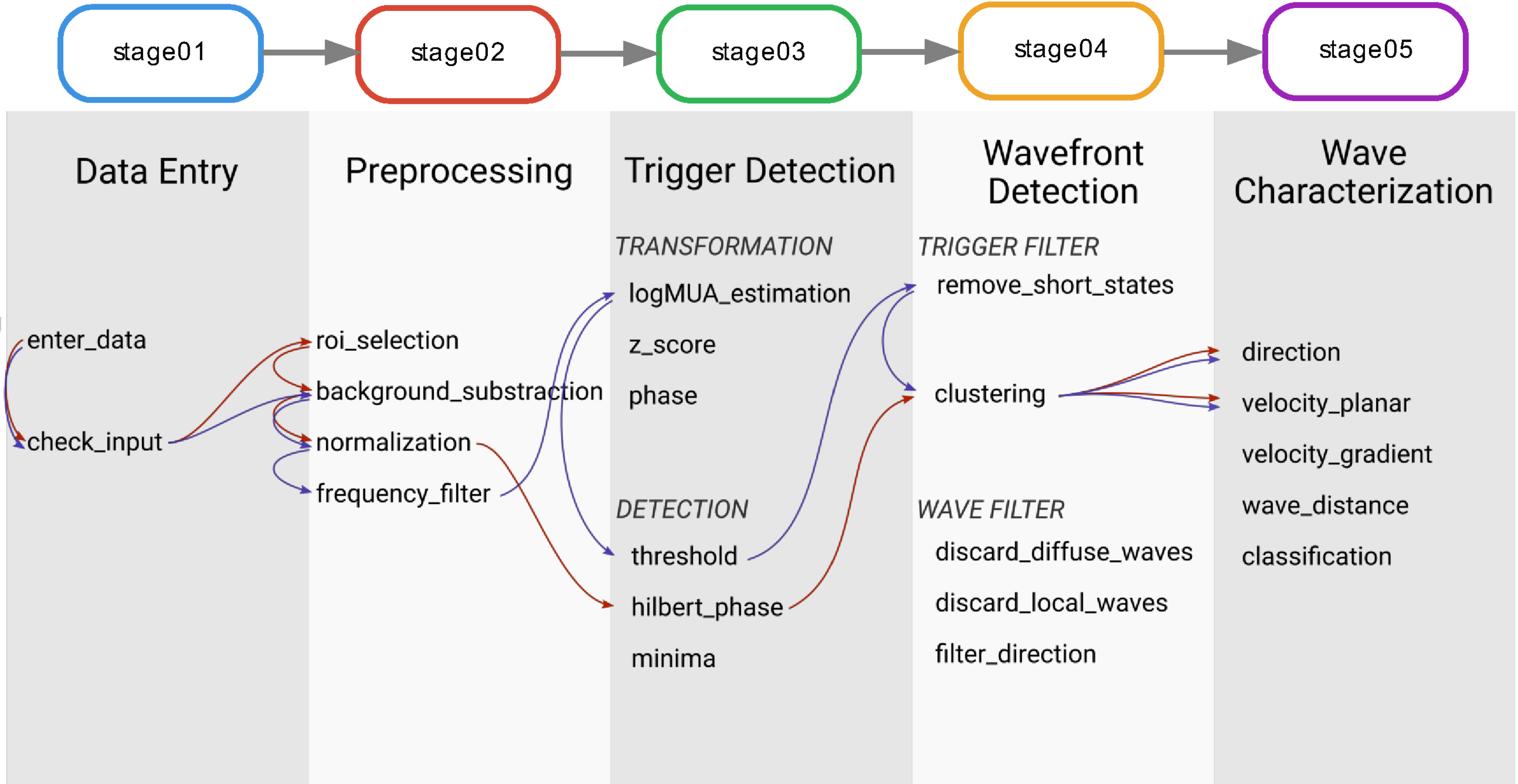
ECoG

EEG

LFP

Spikes

*Simulation*

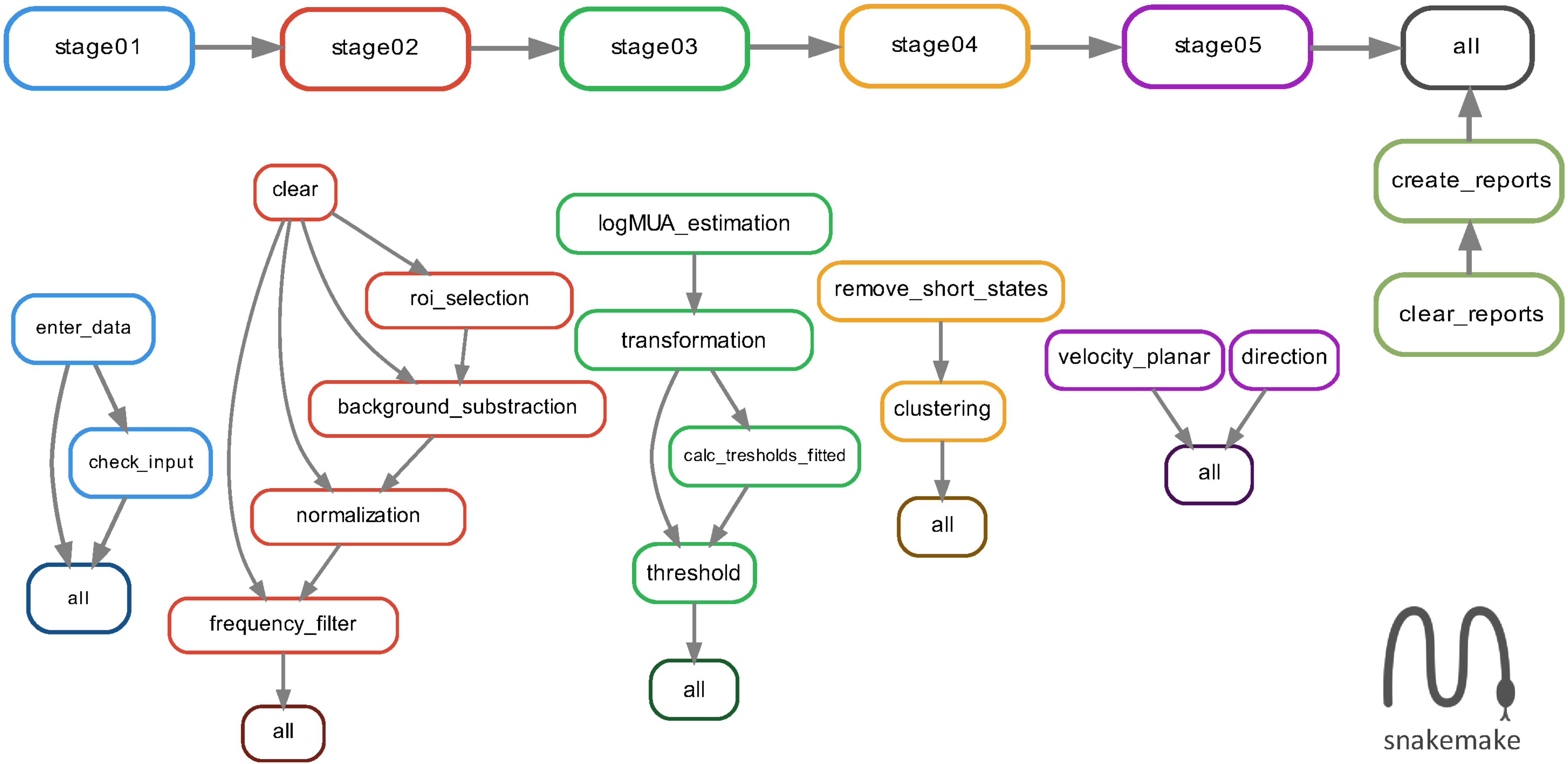



*Data curtesy of: F.S. Pavone Lab, LENS, Florence, Italy / M.V. Sanchez-Vives Lab, IDIBABS, Barcelona, Spain*

Human Brain Project

# Dependency Trees

Pipeline Stage Block



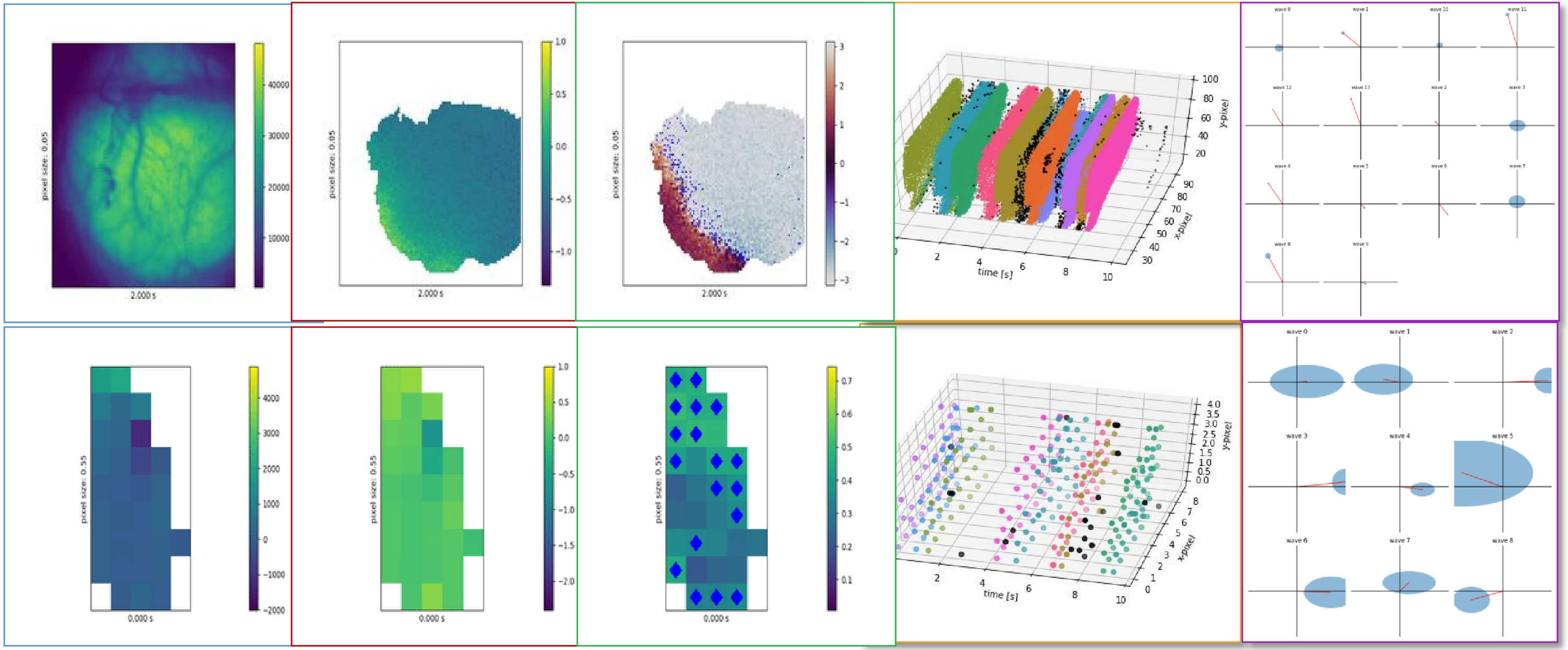

snakemake

Human Brain Project

# Stage Outputs

Pipeline | Stage | Block

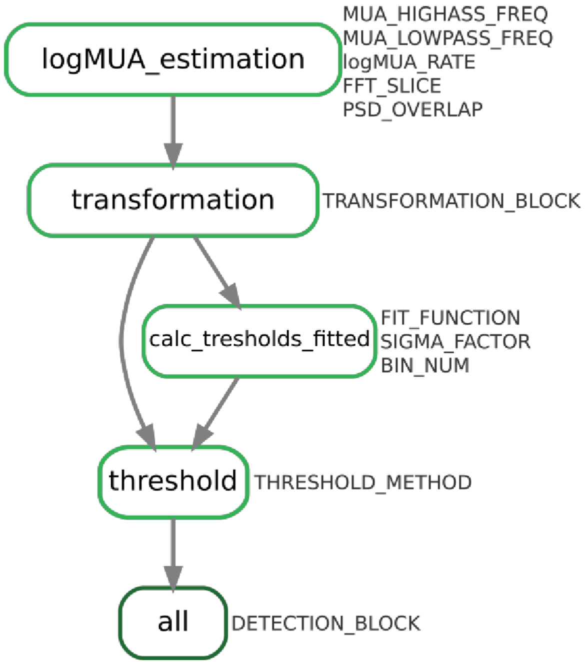
stage01 → stage02 → stage03 → stage04 → stage05



Human Brain Project

# Folder Hierachy

Pipeline | Stage | Block




```
(wavescalephant_env) ✔ ~/pipeline/stage03_trigger_detection
13:54 $ snakemake
```

## Source Files

- stage03_trigger_detection
  - config.yaml
  - Snakefile
  - README.md
  - report.rst
  - scripts
    - transformation
      - z_score.py
      - time_slice.py
      - phase_transform.py
      - logMUA_estimation.py
    - detection
      - calc_thresholds_fixed.py
      - minima.py

## Output Files

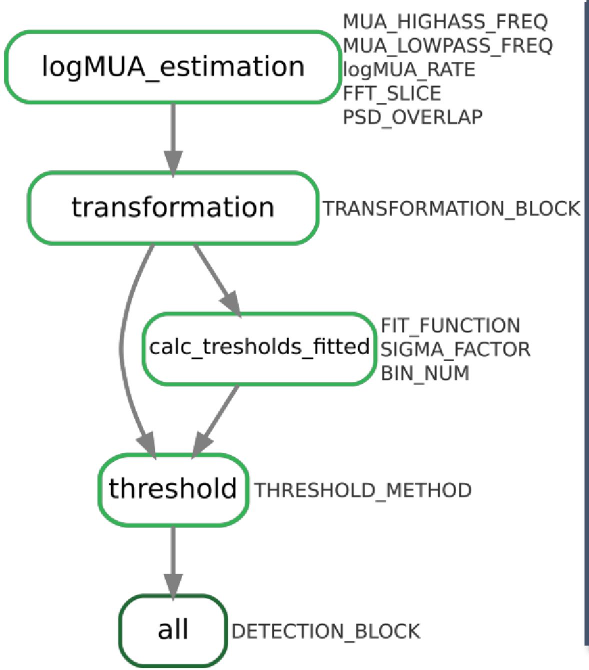
- stage03_trigger_detection
  - threshold
    - amplitude_distribution_channel18.png
    - fitted_thresholds.npy
    - transitions.nix
    - transitions.png
  - transformation
    - logMUA_estimation
      - logMUA_estimation.nix
      - logMUA_trace_1-10s_channel18.png
    - transformed_signal.nix
    - transformed_trace_1-10s_channel18.png
  - trigger_times.nix
  - report.html

Human Brain Project

Pipeline | Stage | Block

# Config & Snakefile

## How to flexibly adapt the trigger detection method?




config.yaml

```
# int or None. default 'None' -> randomly selected
```

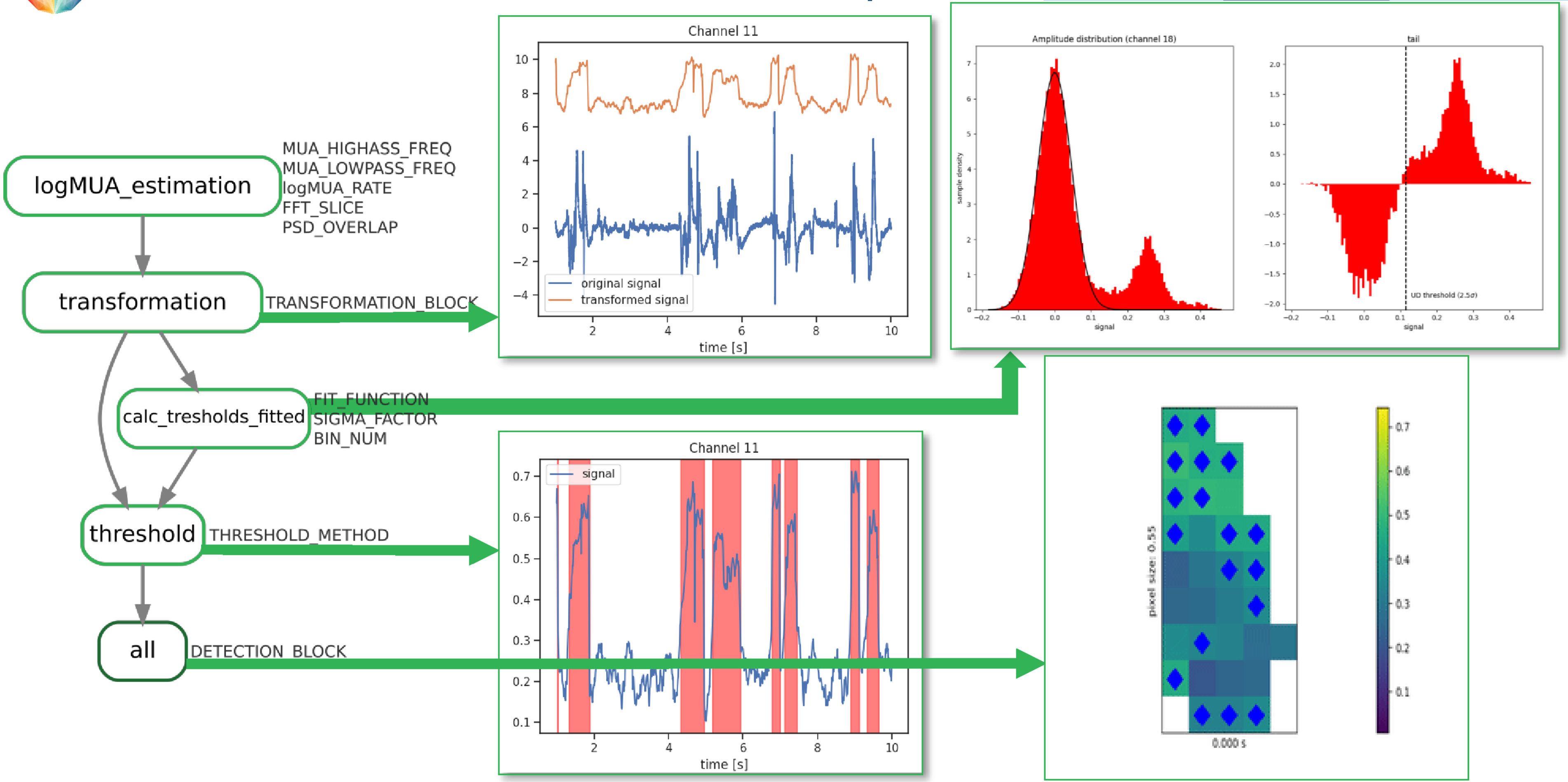
Snakefile

```
    def transformation_input_file(wildcards):
        if wildcards.rule_name == 'transformation':
            return os.path.join(output_path, 'transformation',
                                TRANSFORMATION_BLOCKS[-1],
                                TRANSFORMATION_BLOCKS[-1]+neo_format)
        else:
            idx = locate(TRANSFORMATION_BLOCKS, wildcards.rule_name)
            if idx:
                return os.path.join(output_path, 'transformation',
                                    TRANSFORMATION_BLOCKS[idx-1],
                                    TRANSFORMATION_BLOCKS[idx-1]+neo_format)
            else:
                return stage_input

    def trigger_detection_input_file(wildcards):
        if len(TRANSFORMATION_BLOCKS):
            return os.path.join(output_path, 'transformation',
                                'transformed_signal'+neo_format)
        else:
            return stage_input
```

# Block Outputs





Human Brain Project

# Block Structure



**Block** = Instructions to create output from input (*snakemake rule*)

**Block**

```
rule frequency_filter:
    input:
        data = input_file,
        script = 'scripts/frequency_filter.py',
        config = 'config.yaml'
    output:
        data = os.path.join(output_path, 'frequency_filter.nix'),
    shell:
        """
        python {input.script} --data "{input.data}" \
                              --output "{output.data}" \
                              --highpass_freq {config['HIGHPASS_FREQ']} \
                              --lowpass_freq {config['LOWPASS_FREQ']} \
                              --order {config['FILTER_ORDER']} \
                              --filter_function {config['FILTER_FUNCTION']}
        """
```

elephant
signal_processing.butter

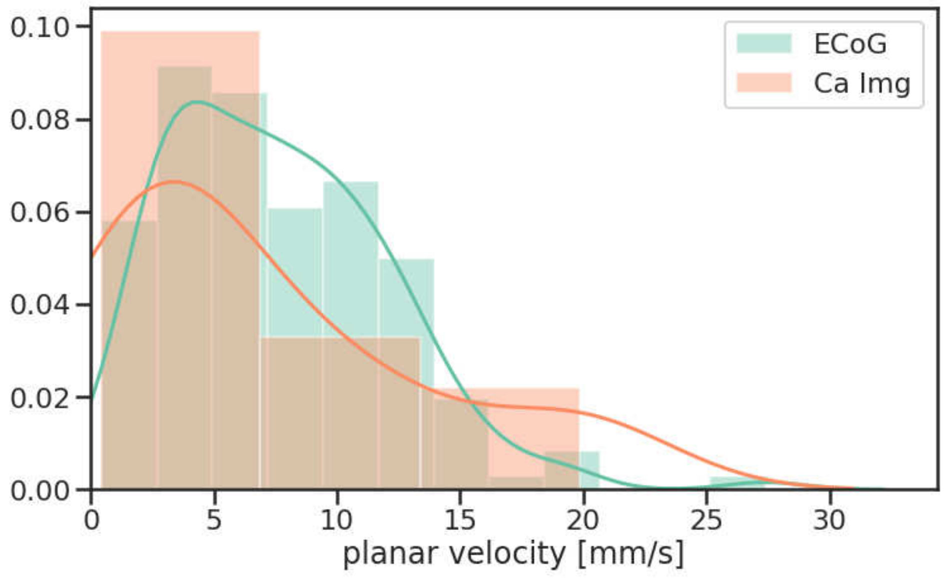
## Snakemake Advantages

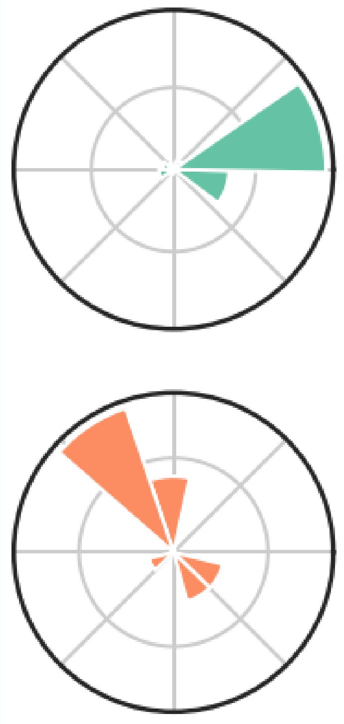
- Can use any shell executables
- Handles portability
- Determines execution order
- Captures execution details

Human Brain Project

# The Payoff

## Direct comparability between different data types

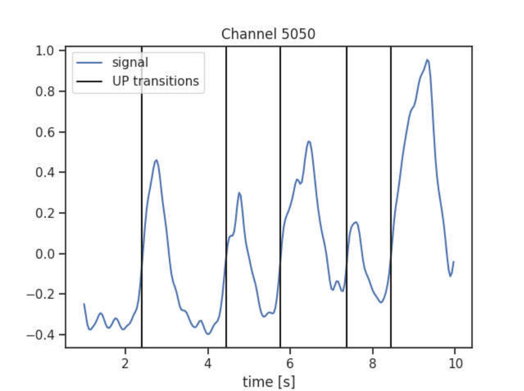


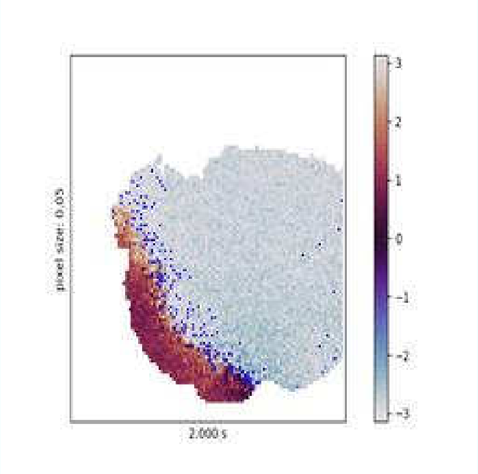



# The Payoff

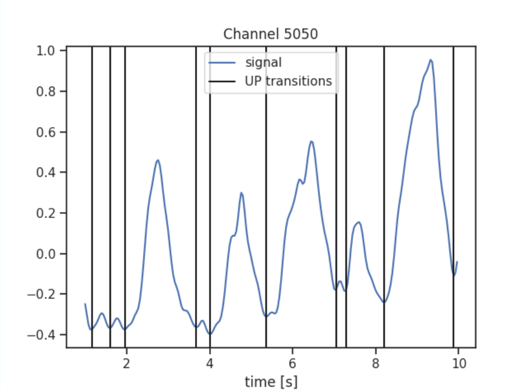
## Benchmarking of analysis algorithms

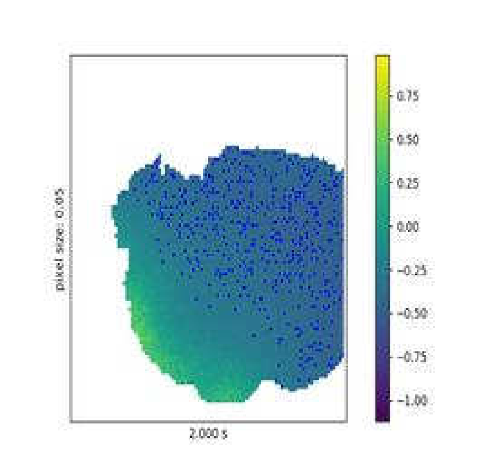
**via Hilbert phase**




**via minima**



# The Payoff

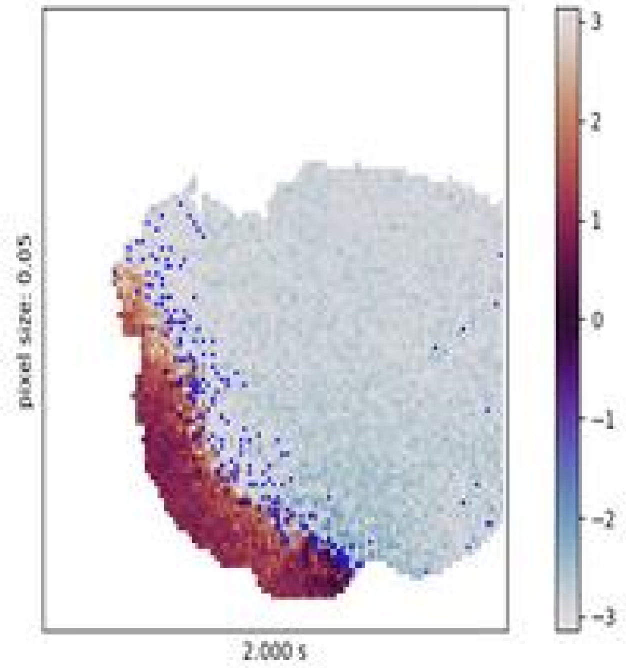
Human Brain Project

## Basis for meaningful validation tests

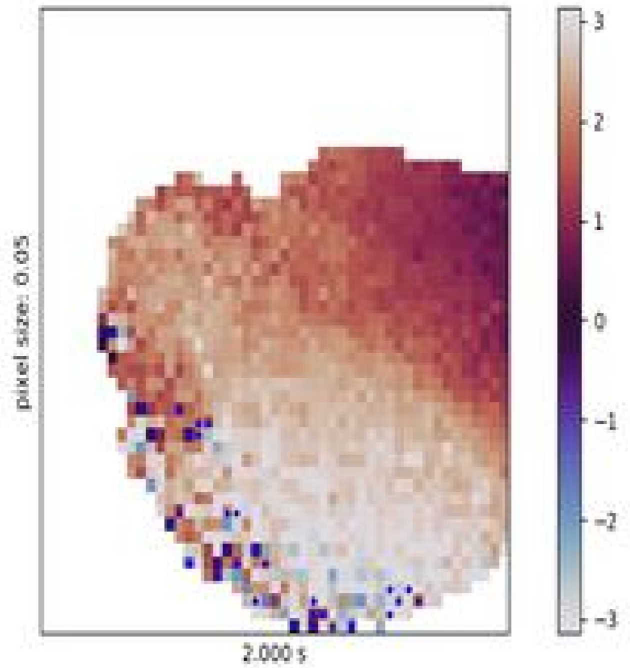
NetworkUnit

SciUnit

recorded activity



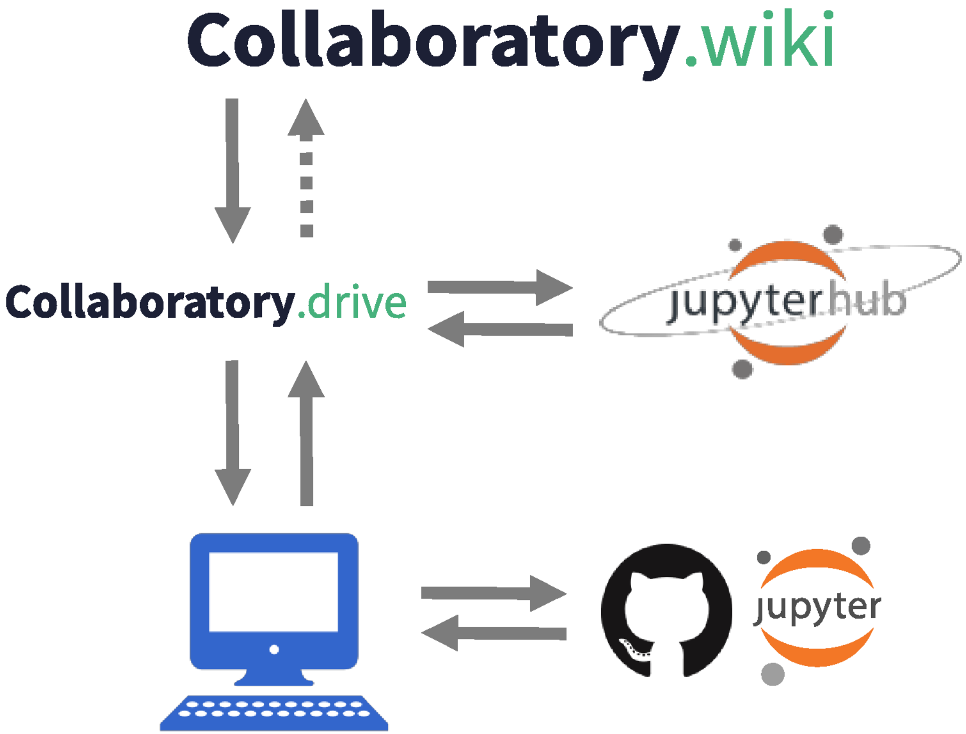

simulated activity




*Model by Cristiano Capone*

Human Brain Project

# Collab Integration




## Wishlist

- GitHub Integration / Version Control
- (Python) Environment Management
- Easy HPC Access
- Workflow Engine

Human Brain Project

# Thank you!

Giulia De Bonis

Cristiano Capone

Elena Pastorelli

Pier Stanislao Paolucci

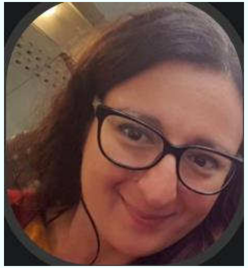
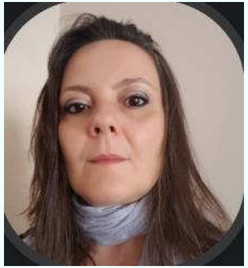
Yann Zerlaut

Glynis Mattheis

Andrew Davison

Robin Gutzen

Michael Denker





JÜLICH Forschungszentrum

INFN Istituto Nazionale di Fisica Nucleare

UNIC

IDIBAPS

lens European Laboratory for Non-Linear Spectroscopy

Università degli Studi Firenze

# THANK YOU!

www.humanbrainproject.eu

@humanbrainproj

@humanbrainproj

HumanBrainProject

Co-funded by
the European Union