SciTrace

A Digital Research Product manager

Nicola Malavasi (LMU Munich)

In collaboration with: Yori Fournier, Kirill Makan, Anastasia Galkin, Olaf Michaelis, Harry Enke

PUNCH Young Academy Tutorial - 11/10/2023

LMU

LUDWIG-
MAXIMILIANS-
UNIVERSITÄT
MÜNCHEN

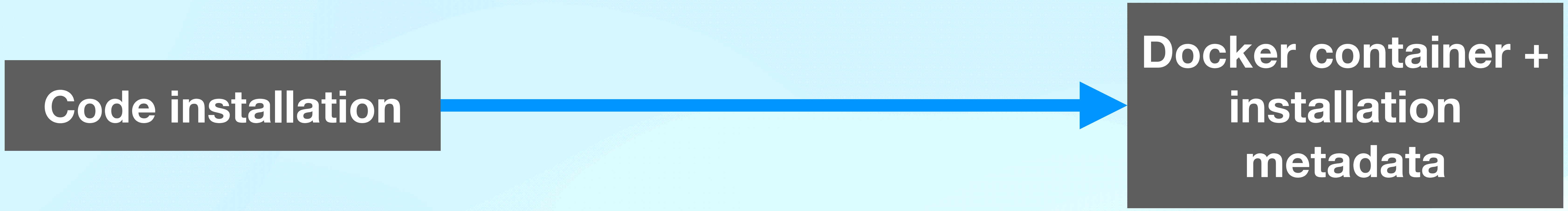
PUNCH
4NFDI

From abstract DRP to real world implementation

Several pieces of software need to be brought together to go from an abstract DRP idea to a prototype. Each piece addresses a specific problem.

From abstract DRP to real world implementation

Several pieces of software need to be brought together to go from an abstract DRP idea to a prototype. Each piece addresses a specific problem.

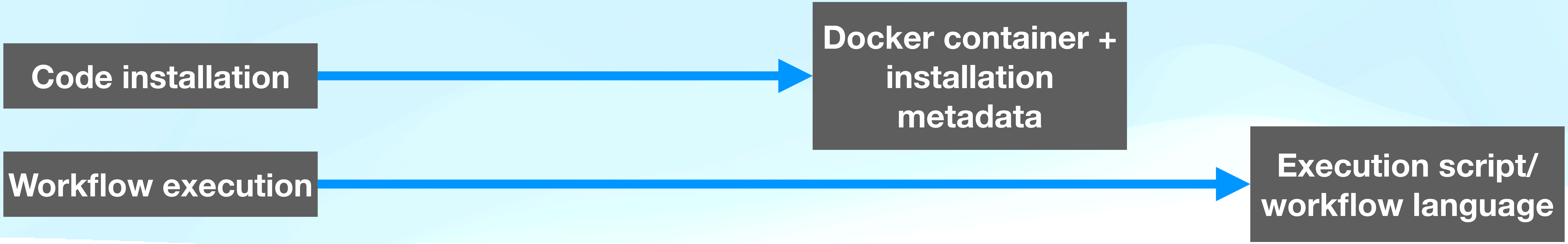


Code installation

Docker container + installation metadata

From abstract DRP to real world implementation

Several pieces of software need to be brought together to go from an abstract DRP idea to a prototype. Each piece addresses a specific problem.



Code installation

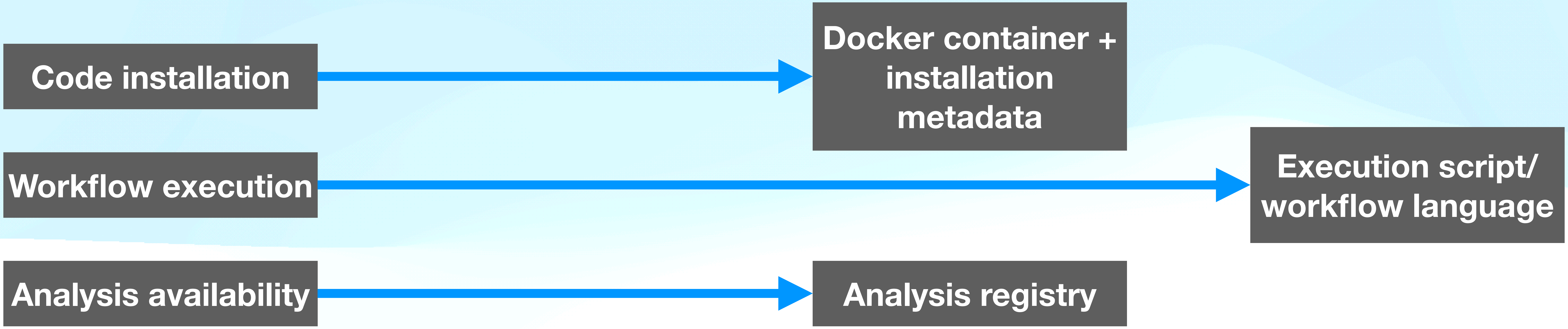
Docker container + installation metadata

Workflow execution

Execution script/ workflow language

From abstract DRP to real world implementation

Several pieces of software need to be brought together to go from an abstract DRP idea to a prototype. Each piece addresses a specific problem.



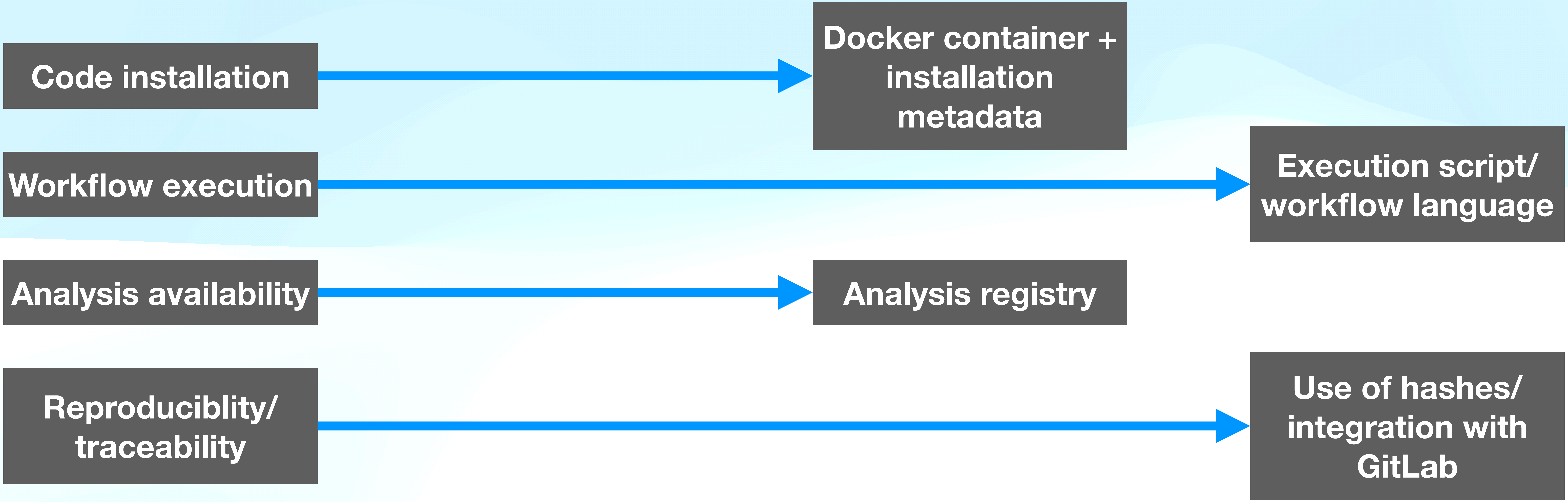
Code installation → Docker container + installation metadata

Workflow execution → Execution script/ workflow language

Analysis availability → Analysis registry

From abstract DRP to real world implementation

Several pieces of software need to be brought together to go from an abstract DRP idea to a prototype. Each piece addresses a specific problem.



Code installation → Docker container + installation metadata

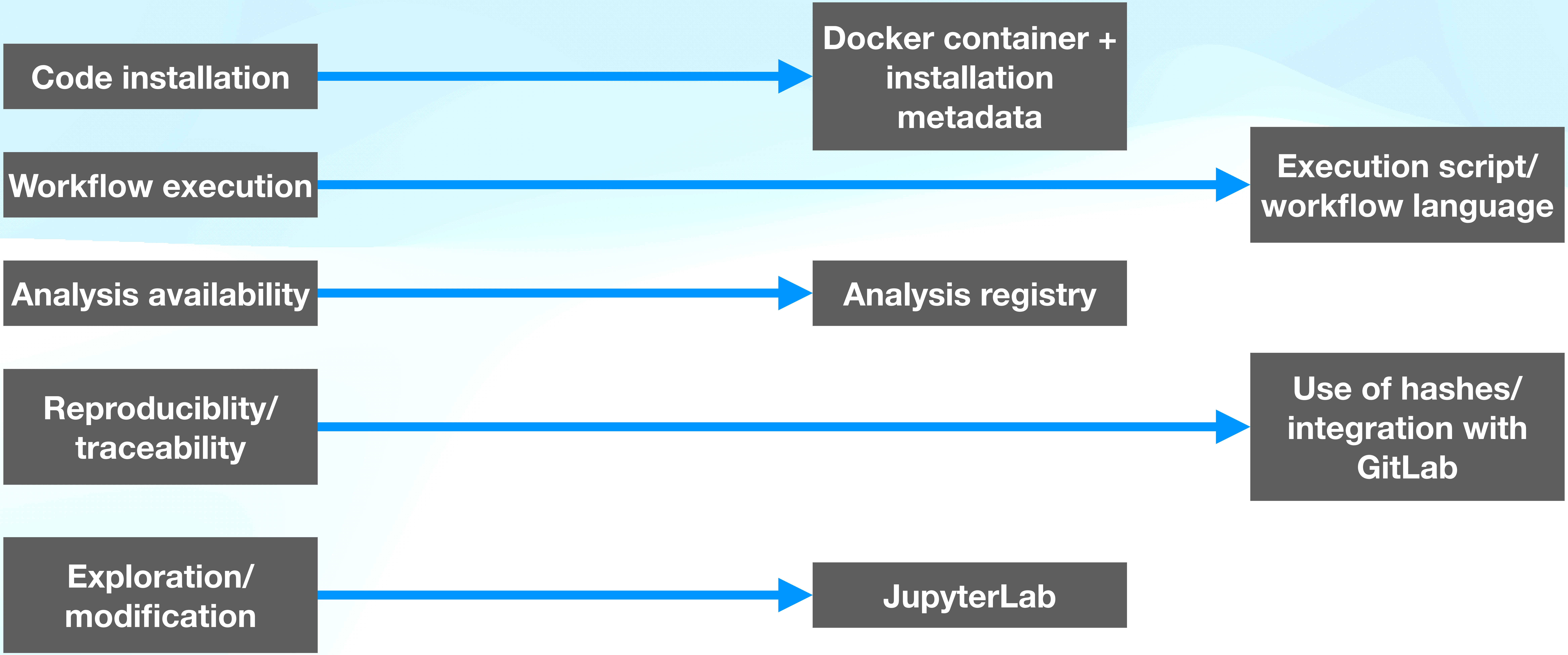
Workflow execution → Execution script/ workflow language

Analysis availability → Analysis registry

Reproduciblity/ traceability → Use of hashes/ integration with GitLab

From abstract DRP to real world implementation

Several pieces of software need to be brought together to go from an abstract DRP idea to a prototype. Each piece addresses a specific problem.



Code installation

Docker container + installation metadata

Workflow execution

Execution script/ workflow language

Analysis availability

Analysis registry

Reproduciblity/ traceability

Use of hashes/ integration with GitLab

Exploration/ modification

JupyterLab

SciTrace

Having established the abstract concept of a DRP and a check list of software needed to build a prototype, we can move forward.

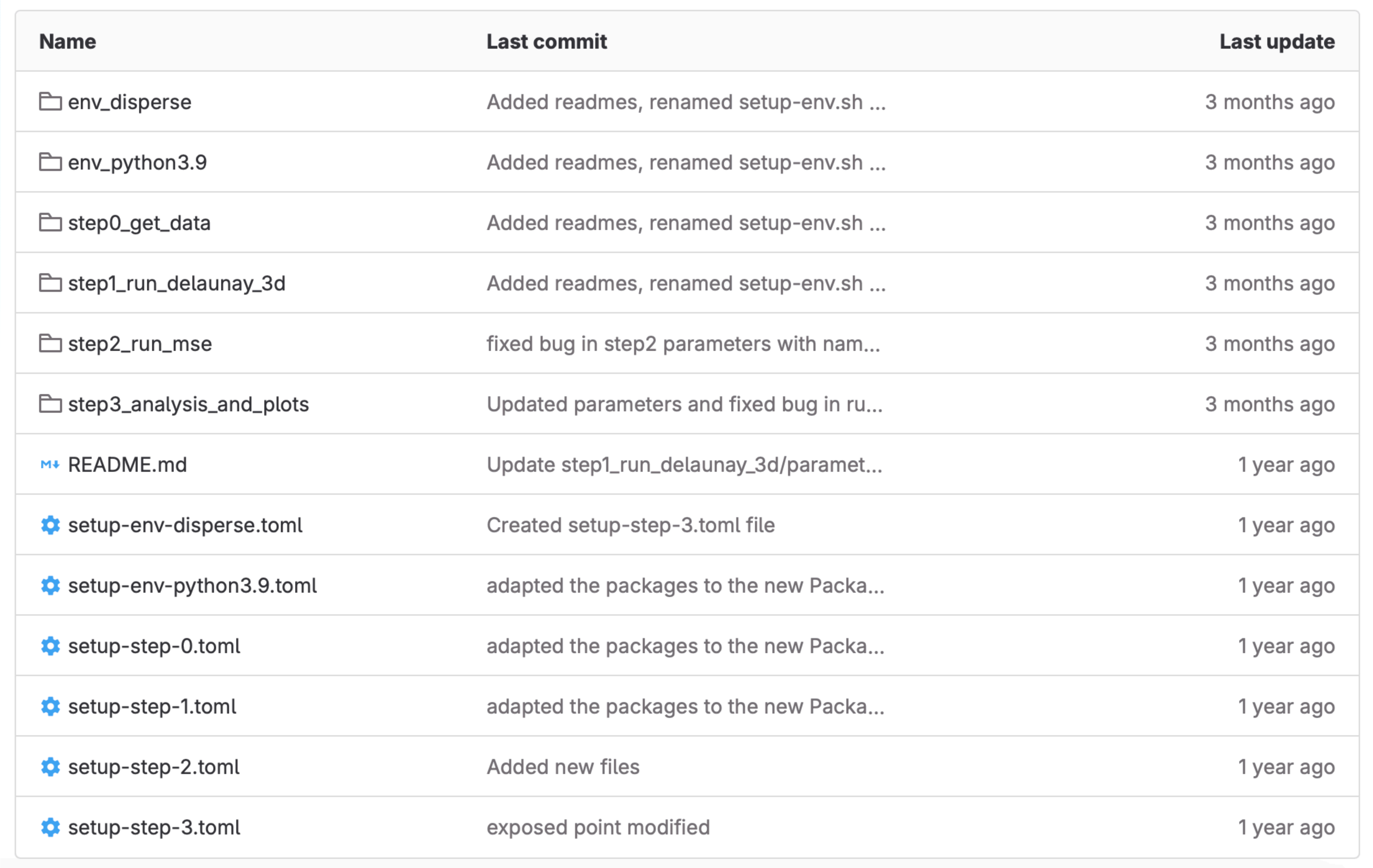
Prototype software created at AIP Potsdam by Yori Fournier, Kirill Makan, Anastasia Galkin, Olaf Michaelis: SciTrace

The DRP in the SciTrace formalism

Necessary elements

In the SciTrace formalism, a DRP is a package formed by a structure hosted in an existing GitLab repository.

This repository can be ingested by the SciTrace program, that can use its parts to create the DRP.

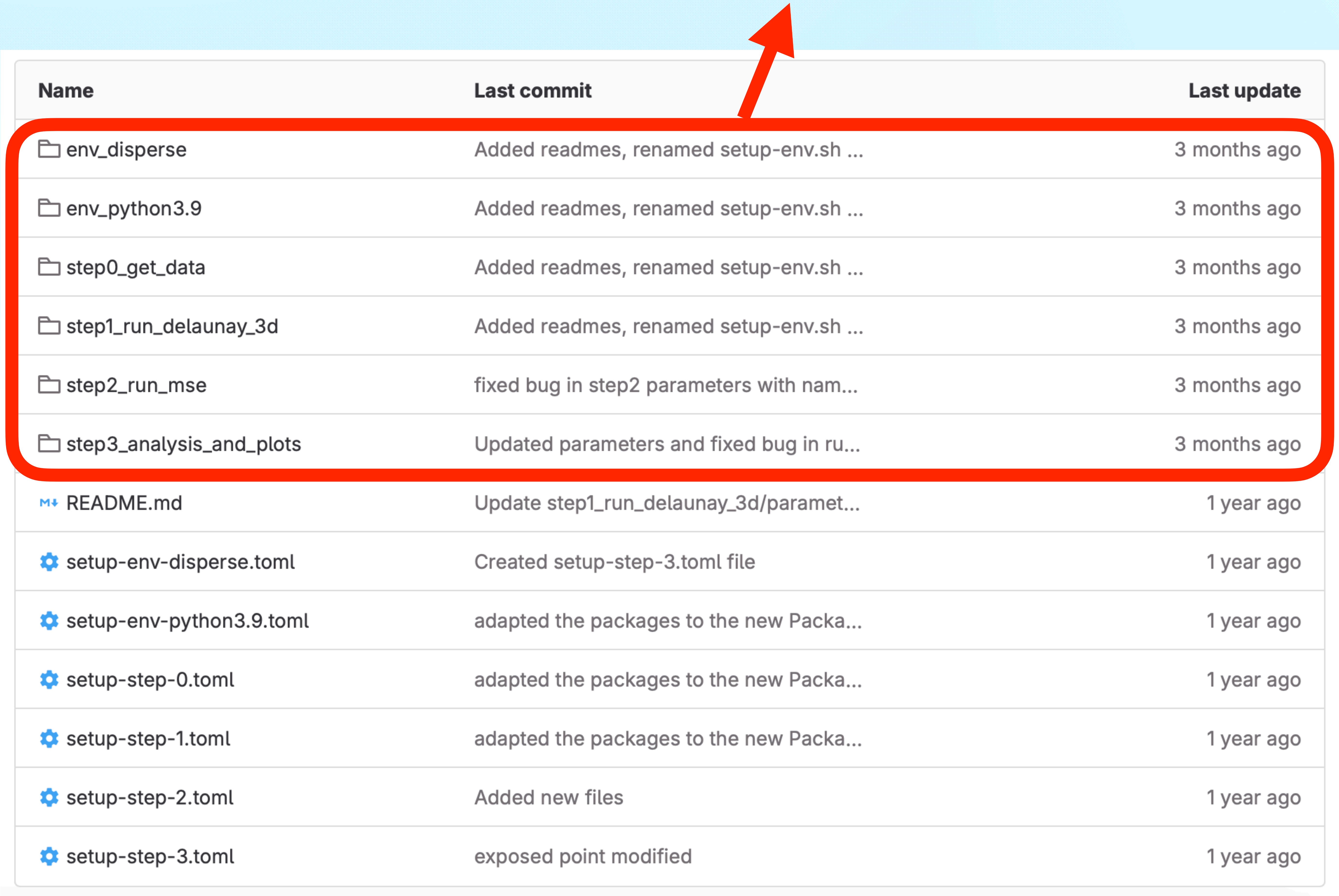


Name	Last commit	Last update
env_disperse	Added readmes, renamed setup-env.sh ...	3 months ago
env_python3.9	Added readmes, renamed setup-env.sh ...	3 months ago
step0_get_data	Added readmes, renamed setup-env.sh ...	3 months ago
step1_run_delaunay_3d	Added readmes, renamed setup-env.sh ...	3 months ago
step2_run_mse	fixed bug in step2 parameters with nam...	3 months ago
step3_analysis_and_plots	Updated parameters and fixed bug in ru...	3 months ago
README.md	Update step1_run_delaunay_3d/paramet...	1 year ago
setup-env-disperse.toml	Created setup-step-3.toml file	1 year ago
setup-env-python3.9.toml	adapted the packages to the new Packa...	1 year ago
setup-step-0.toml	adapted the packages to the new Packa...	1 year ago
setup-step-1.toml	adapted the packages to the new Packa...	1 year ago
setup-step-2.toml	Added new files	1 year ago
setup-step-3.toml	exposed point modified	1 year ago

The DRP in the SciTrace formalism

Necessary elements

Each of these is a separate DRP.



Name	Last commit	Last update
env_disperse	Added readmes, renamed setup-env.sh ...	3 months ago
env_python3.9	Added readmes, renamed setup-env.sh ...	3 months ago
step0_get_data	Added readmes, renamed setup-env.sh ...	3 months ago
step1_run_delaunay_3d	Added readmes, renamed setup-env.sh ...	3 months ago
step2_run_mse	fixed bug in step2 parameters with nam...	3 months ago
step3_analysis_and_plots	Updated parameters and fixed bug in ru...	3 months ago
README.md	Update step1_run_delaunay_3d/paramet...	1 year ago
setup-env-disperse.toml	Created setup-step-3.toml file	1 year ago
setup-env-python3.9.toml	adapted the packages to the new Packa...	1 year ago
setup-step-0.toml	adapted the packages to the new Packa...	1 year ago
setup-step-1.toml	adapted the packages to the new Packa...	1 year ago
setup-step-2.toml	Added new files	1 year ago
setup-step-3.toml	exposed point modified	1 year ago

In the SciTrace formalism, a DRP is a package formed by a structure hosted in an existing GitLab repository.

This repository can be ingested by the SciTrace program, that can use its parts to create the DRP.

Package repository

Necessary requirements

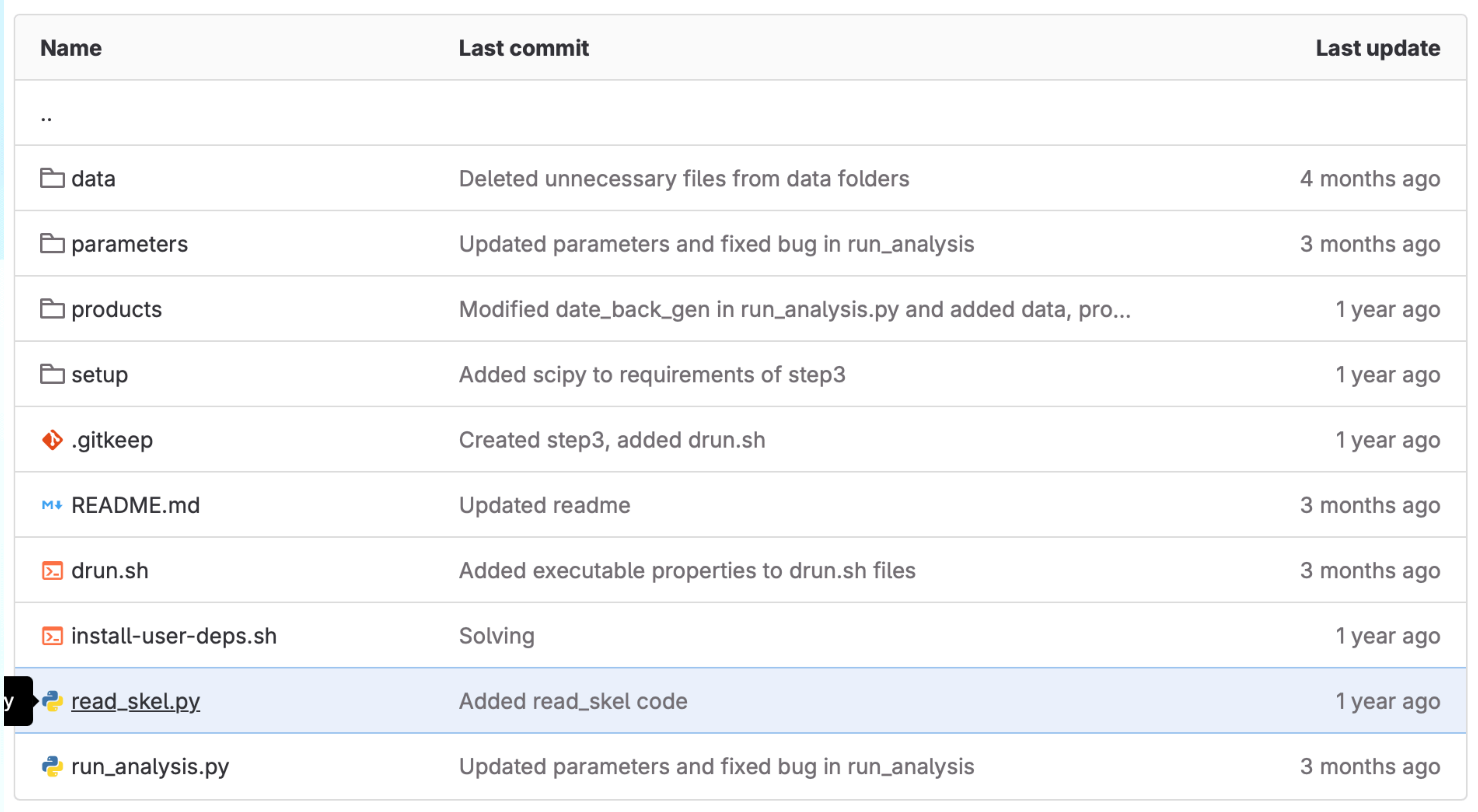
- Data folder: contains the input
- Parameters folder: contains the parameters for the analysis (toml file)
- Products folder: will contain the output
- Execution script: drun.sh
- Analysis scripts
- Installation scripts: install.sh, install-deps.sh, and install-user-deps.sh

Package folder

Necessary requirements

Data folder Parameters folder Products folder Execution script: drun.sh Analysis scripts

Installation scripts: install.sh, install-deps.sh, and install-user-deps.sh



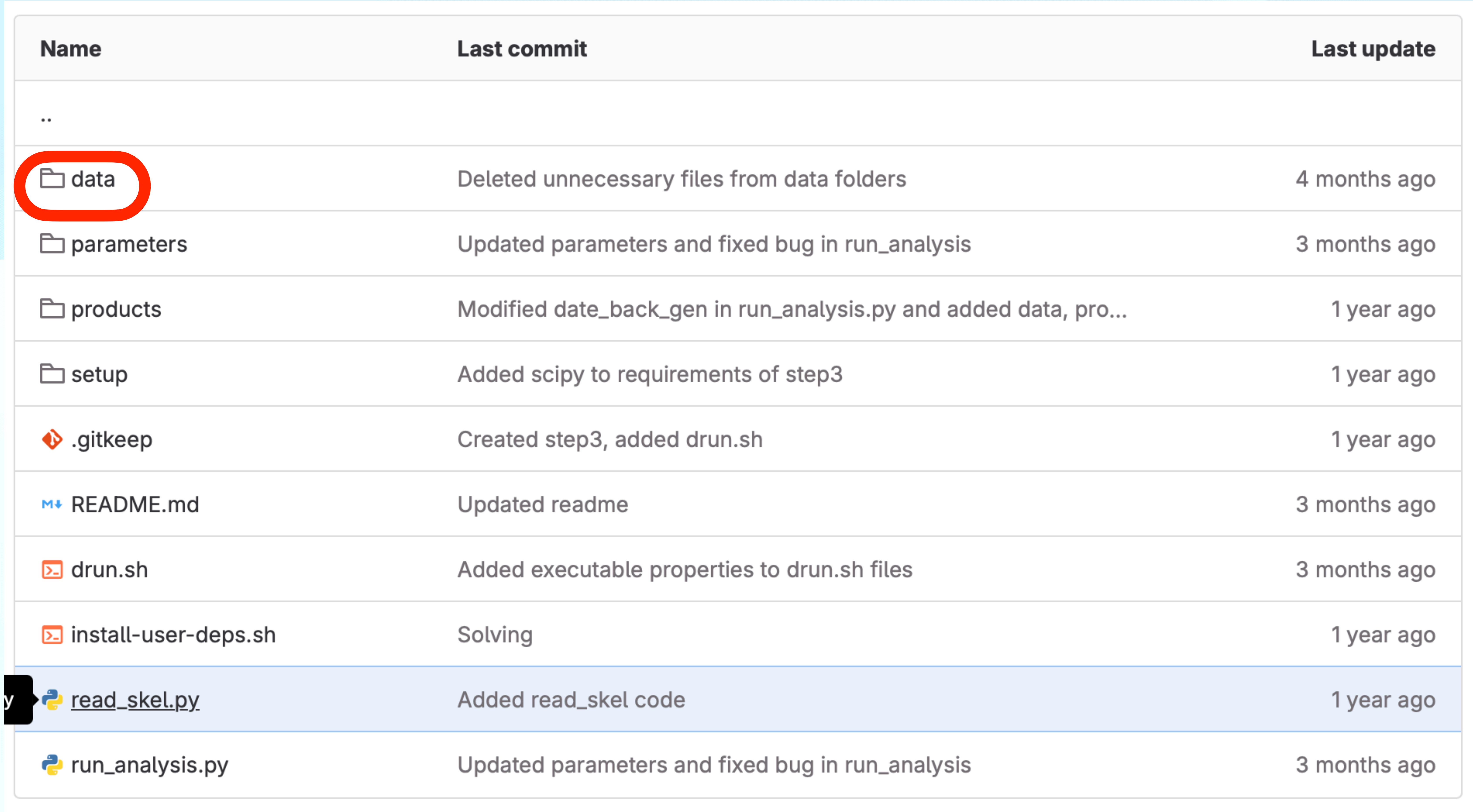
Name	Last commit	Last update
..		
data	Deleted unnecessary files from data folders	4 months ago
parameters	Updated parameters and fixed bug in run_analysis	3 months ago
products	Modified date_back_gen in run_analysis.py and added data, pro...	1 year ago
setup	Added scipy to requirements of step3	1 year ago
.gitkeep	Created step3, added drun.sh	1 year ago
README.md	Updated readme	3 months ago
drun.sh	Added executable properties to drun.sh files	3 months ago
install-user-deps.sh	Solving	1 year ago
read_skel.py	Added read_skel code	1 year ago
run_analysis.py	Updated parameters and fixed bug in run_analysis	3 months ago

Package folder

Necessary requirements

Data folder Parameters folder Products folder Execution script: drun.sh Analysis scripts

Installation scripts: install.sh, install-deps.sh, and install-user-deps.sh



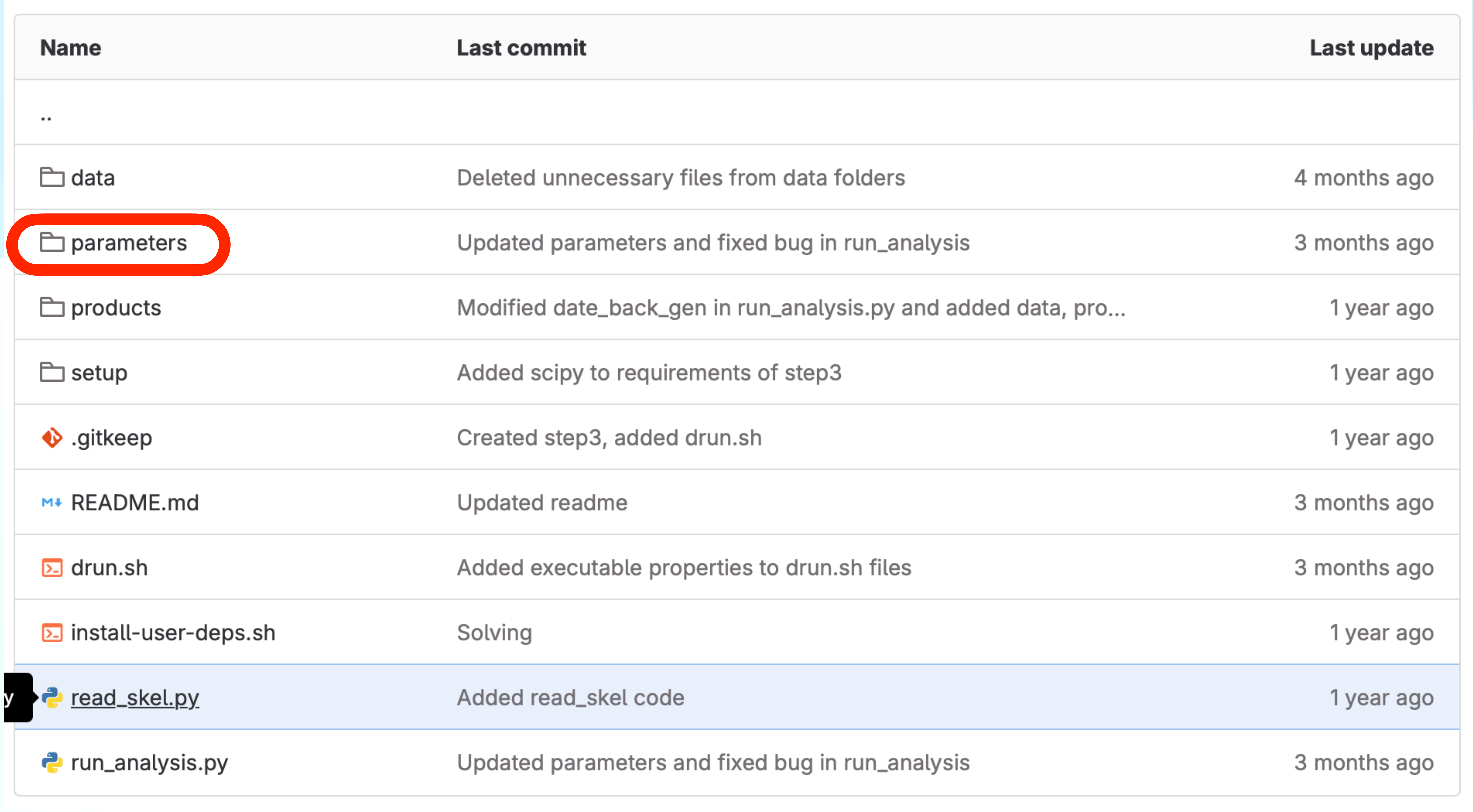
Name	Last commit	Last update
..		
data	Deleted unnecessary files from data folders	4 months ago
parameters	Updated parameters and fixed bug in run_analysis	3 months ago
products	Modified date_back_gen in run_analysis.py and added data, pro...	1 year ago
setup	Added scipy to requirements of step3	1 year ago
.gitkeep	Created step3, added drun.sh	1 year ago
README.md	Updated readme	3 months ago
drun.sh	Added executable properties to drun.sh files	3 months ago
install-user-deps.sh	Solving	1 year ago
read_skel.py	Added read_skel code	1 year ago
run_analysis.py	Updated parameters and fixed bug in run_analysis	3 months ago

Package folder

Necessary requirements

Data folder Parameters folder Products folder Execution script: drun.sh Analysis scripts

Installation scripts: install.sh, install-deps.sh, and install-user-deps.sh



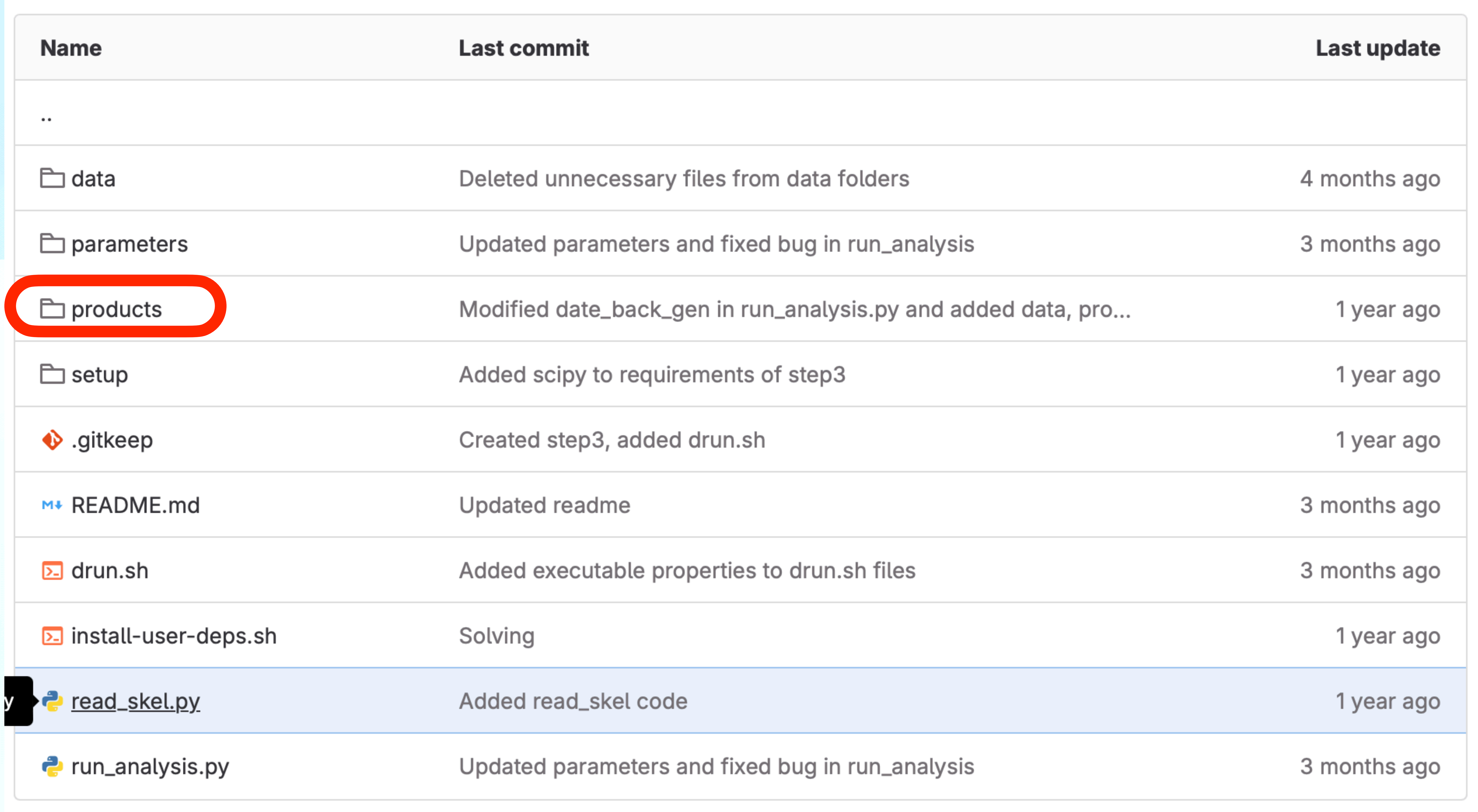
Name	Last commit	Last update
..		
data	Deleted unnecessary files from data folders	4 months ago
parameters	Updated parameters and fixed bug in run_analysis	3 months ago
products	Modified date_back_gen in run_analysis.py and added data, pro...	1 year ago
setup	Added scipy to requirements of step3	1 year ago
.gitkeep	Created step3, added drun.sh	1 year ago
README.md	Updated readme	3 months ago
drun.sh	Added executable properties to drun.sh files	3 months ago
install-user-deps.sh	Solving	1 year ago
read_skel.py	Added read_skel code	1 year ago
run_analysis.py	Updated parameters and fixed bug in run_analysis	3 months ago

Package folder

Necessary requirements

Data folder Parameters folder Products folder Execution script: drun.sh Analysis scripts

Installation scripts: install.sh, install-deps.sh, and install-user-deps.sh



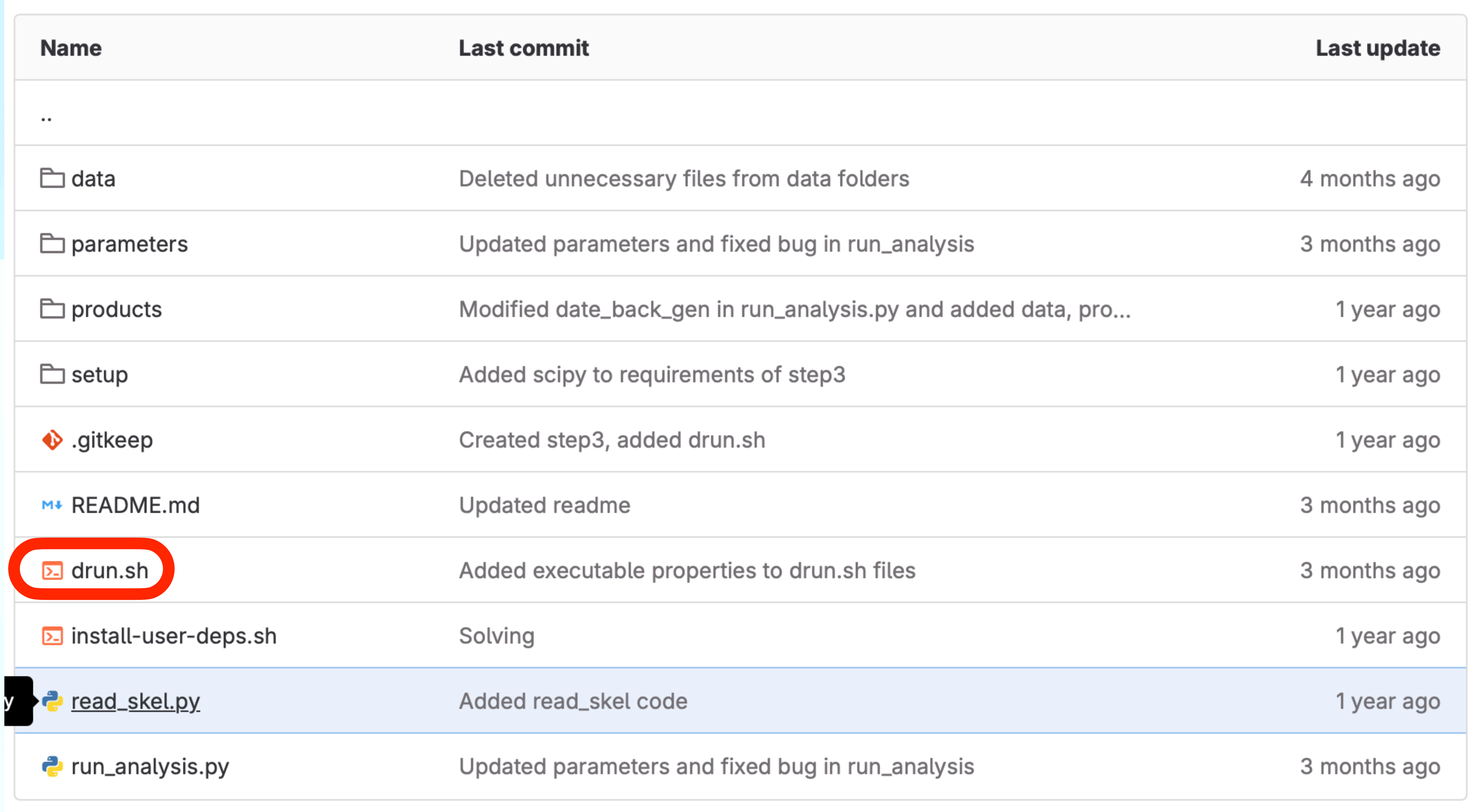
Name	Last commit	Last update
..		
data	Deleted unnecessary files from data folders	4 months ago
parameters	Updated parameters and fixed bug in run_analysis	3 months ago
products	Modified date_back_gen in run_analysis.py and added data, pro...	1 year ago
setup	Added scipy to requirements of step3	1 year ago
.gitkeep	Created step3, added drun.sh	1 year ago
README.md	Updated readme	3 months ago
drun.sh	Added executable properties to drun.sh files	3 months ago
install-user-deps.sh	Solving	1 year ago
read_skel.py	Added read_skel code	1 year ago
run_analysis.py	Updated parameters and fixed bug in run_analysis	3 months ago

Package folder

Necessary requirements

Data folder Parameters folder Products folder Execution script: drun.sh Analysis scripts

Installation scripts: install.sh, install-deps.sh, and install-user-deps.sh



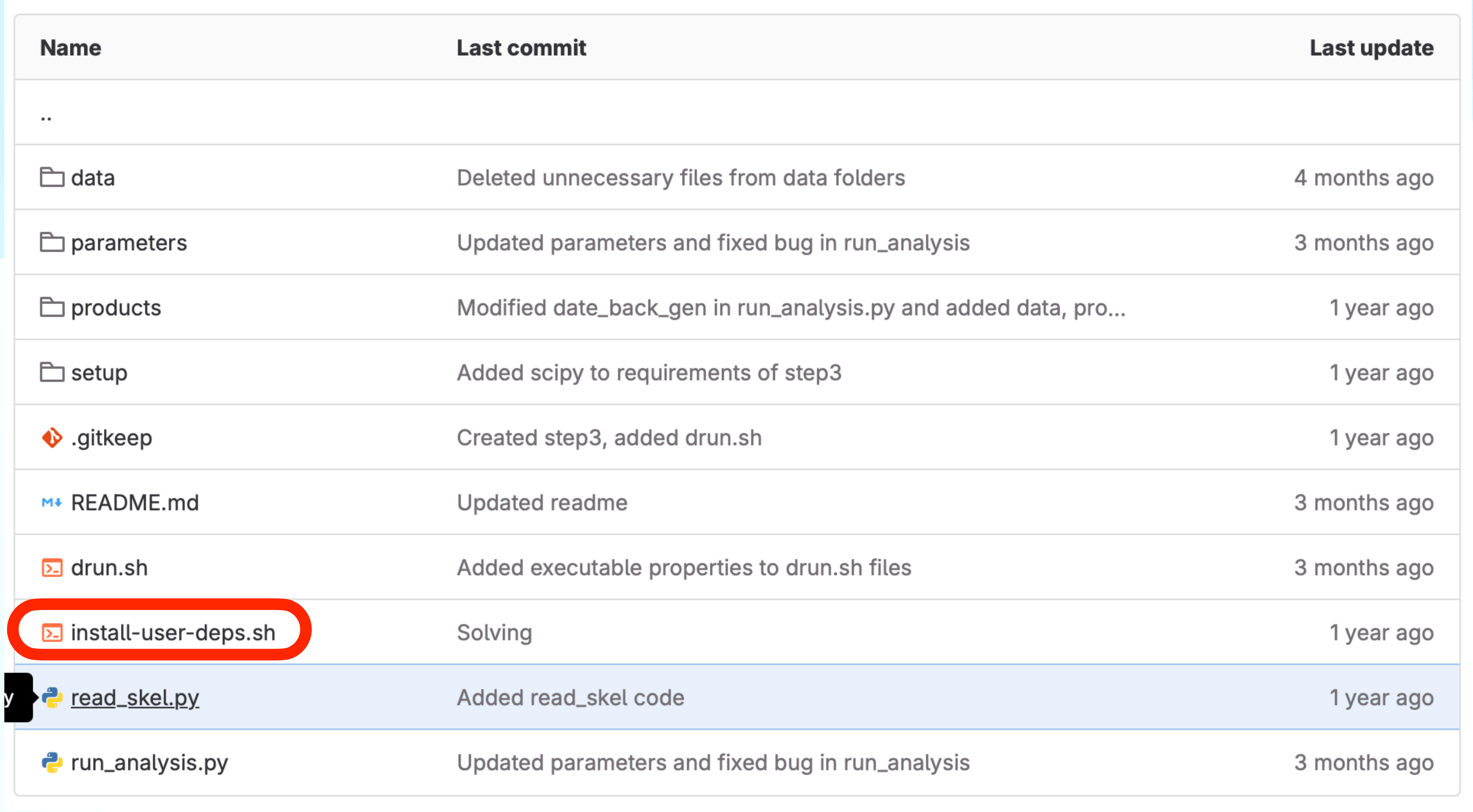
Name	Last commit	Last update
..		
data	Deleted unnecessary files from data folders	4 months ago
parameters	Updated parameters and fixed bug in run_analysis	3 months ago
products	Modified date_back_gen in run_analysis.py and added data, pro...	1 year ago
setup	Added scipy to requirements of step3	1 year ago
.gitkeep	Created step3, added drun.sh	1 year ago
README.md	Updated readme	3 months ago
drun.sh	Added executable properties to drun.sh files	3 months ago
install-user-deps.sh	Solving	1 year ago
read_skel.py	Added read_skel code	1 year ago
run_analysis.py	Updated parameters and fixed bug in run_analysis	3 months ago

Package folder

Necessary requirements

Data folder Parameters folder Products folder Execution script: drun.sh Analysis scripts

Installation scripts: install.sh, install-deps.sh, and install-user-deps.sh



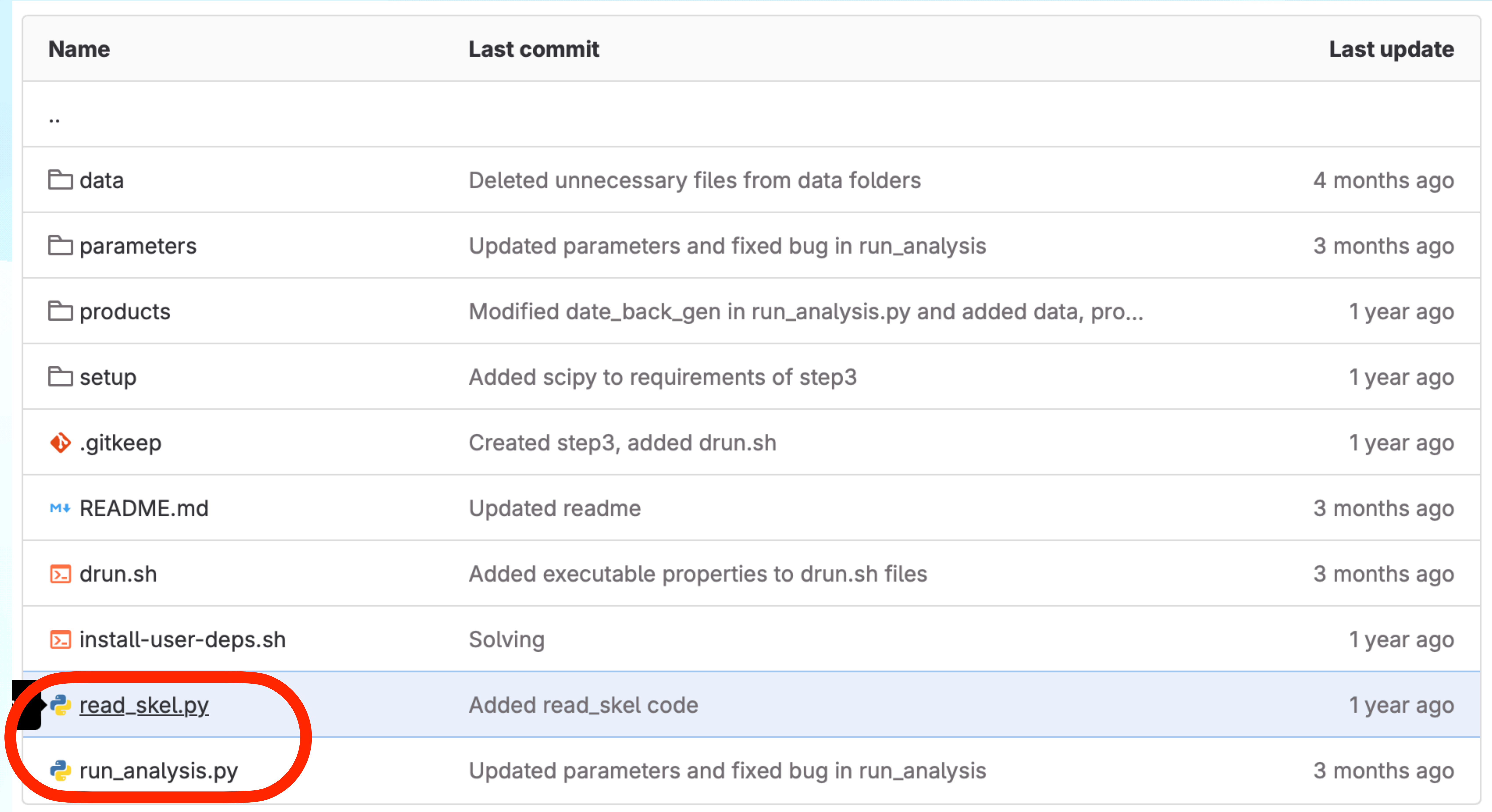
Name	Last commit	Last update
..		
data	Deleted unnecessary files from data folders	4 months ago
parameters	Updated parameters and fixed bug in run_analysis	3 months ago
products	Modified date_back_gen in run_analysis.py and added data, pro...	1 year ago
setup	Added scipy to requirements of step3	1 year ago
.gitkeep	Created step3, added drun.sh	1 year ago
README.md	Updated readme	3 months ago
drun.sh	Added executable properties to drun.sh files	3 months ago
install-user-deps.sh	Solving	1 year ago
read_skel.py	Added read_skel code	1 year ago
run_analysis.py	Updated parameters and fixed bug in run_analysis	3 months ago

Package folder

Necessary requirements

Data folder Parameters folder Products folder Execution script: drun.sh Analysis scripts

Installation scripts: install.sh, install-deps.sh, and install-user-deps.sh



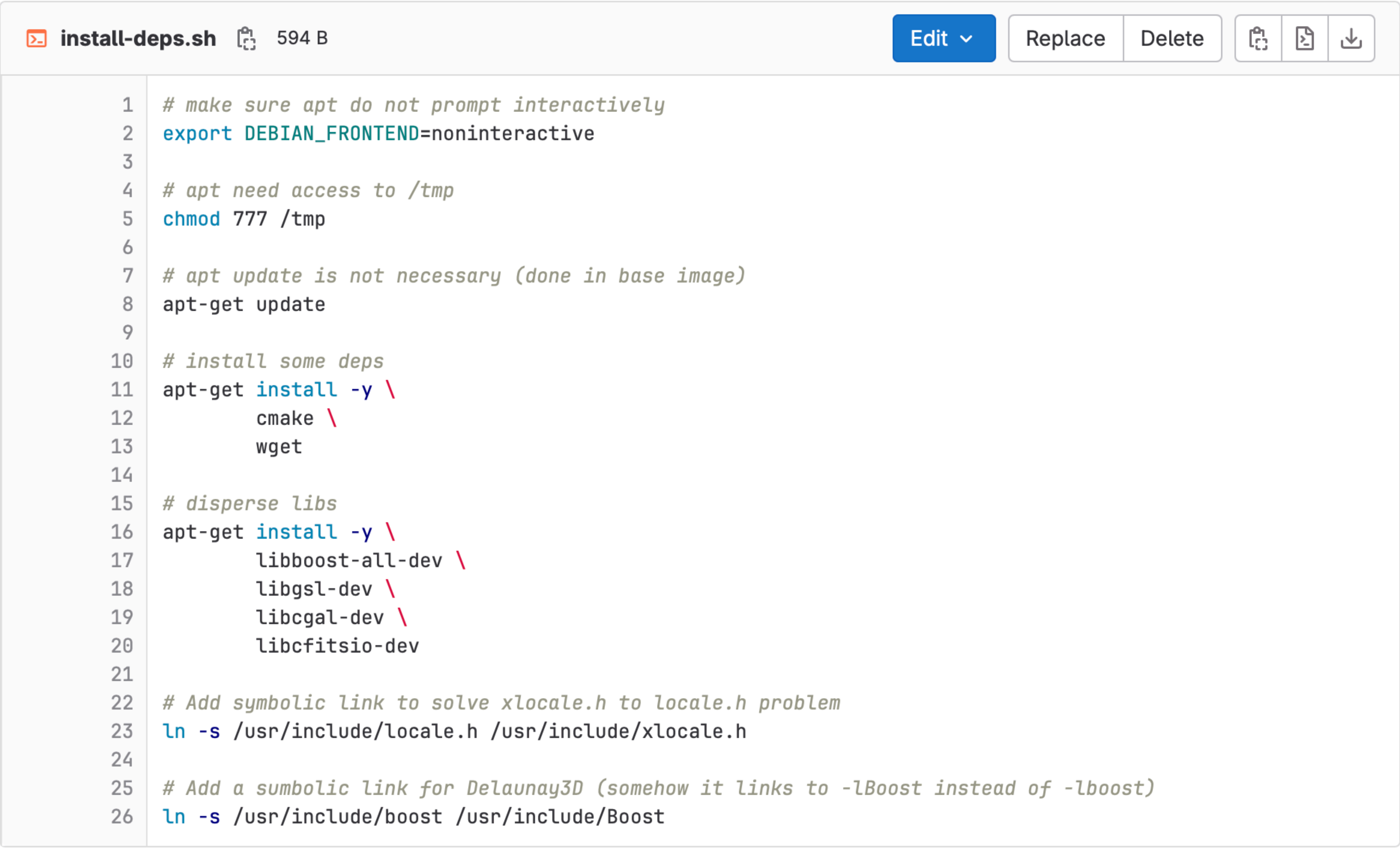
Name	Last commit	Last update
..		
data	Deleted unnecessary files from data folders	4 months ago
parameters	Updated parameters and fixed bug in run_analysis	3 months ago
products	Modified date_back_gen in run_analysis.py and added data, pro...	1 year ago
setup	Added scipy to requirements of step3	1 year ago
.gitkeep	Created step3, added drun.sh	1 year ago
README.md	Updated readme	3 months ago
drun.sh	Added executable properties to drun.sh files	3 months ago
install-user-deps.sh	Solving	1 year ago
read_skel.py	Added read_skel code	1 year ago
run_analysis.py	Updated parameters and fixed bug in run_analysis	3 months ago

Installation

install-deps.sh

This script installs the dependencies necessary for the code.

Treated as sequence of bash commands, it is transformed in an ad-hoc Dockerfile to generate a container.



```bash
# make sure apt do not prompt interactively
export DEBIAN_FRONTEND=noninteractive

# apt need access to /tmp
chmod 777 /tmp

# apt update is not necessary (done in base image)
apt-get update

# install some deps
apt-get install -y \
        cmake \
        wget

# disperse libs
apt-get install -y \
        libboost-all-dev \
        libgsl-dev \
        libcgal-dev \
        libcfitsio-dev

# Add symbolic link to solve xlocale.h to locale.h problem
ln -s /usr/include/locale.h /usr/include/xlocale.h

# Add a sumbolic link for Delaunay3D (somehow it links to -lBoost instead of -lboost)
ln -s /usr/include/boost /usr/include/Boost
```

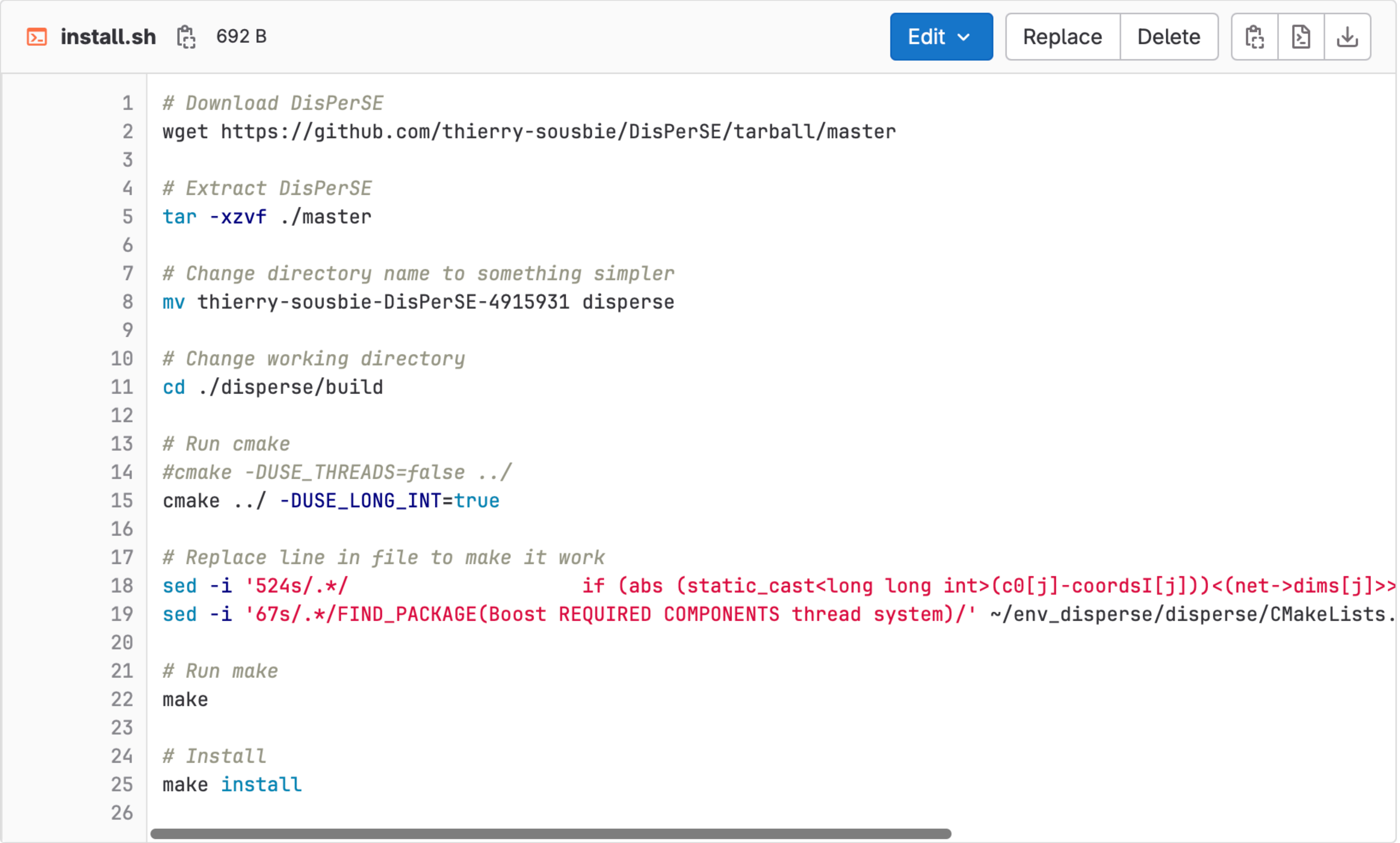
Example of install-deps.sh for DisPerSE

Installation

install.sh

This script installs the necessary code.

Treated as sequence of bash commands, it is transformed in an ad-hoc Dockerfile to generate a container.



```
# Download DisPerSE
wget https://github.com/thierry-sousbie/DisPerSE/tarball/master

# Extract DisPerSE
tar -xzvf ./master

# Change directory name to something simpler
mv thierry-sousbie-DisPerSE-4915931 disperse

# Change working directory
cd ./disperse/build

# Run cmake
#cmake -DUSE_THREADS=false ../
cmake ../ -DUSE_LONG_INT=true

# Replace line in file to make it work
sed -i '524s/.*/                    if (abs (static_cast<long long int>(c0[j]-coordsI[j]))<(net->dims[j]>>
sed -i '67s/.*/FIND_PACKAGE(Boost REQUIRED COMPONENTS thread system)/' ~/env_disperse/disperse/CMakeLists.

# Run make
make

# Install
make install

```

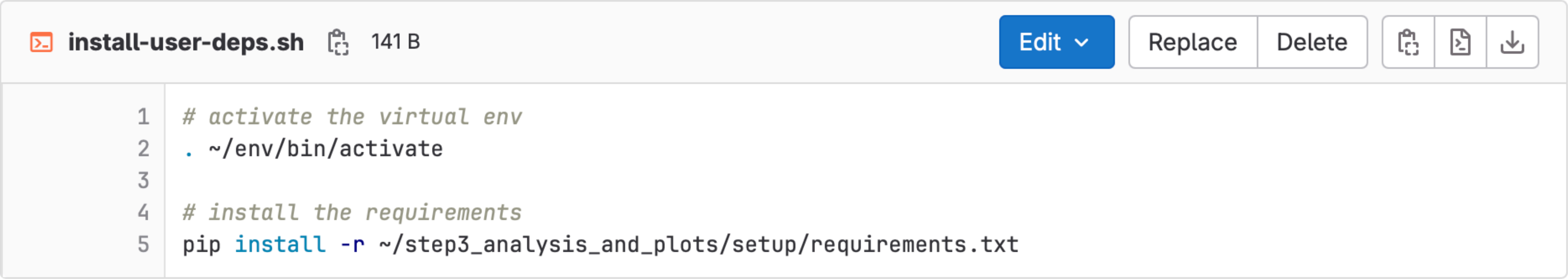
Example of install.sh for DisPerSE

Installation

install-user-deps.sh

This script activates the python environment for exploration and installs the necessary code.

Treated as sequence of bash commands, it is transformed in an ad-hoc Dockerfile to generate a container.

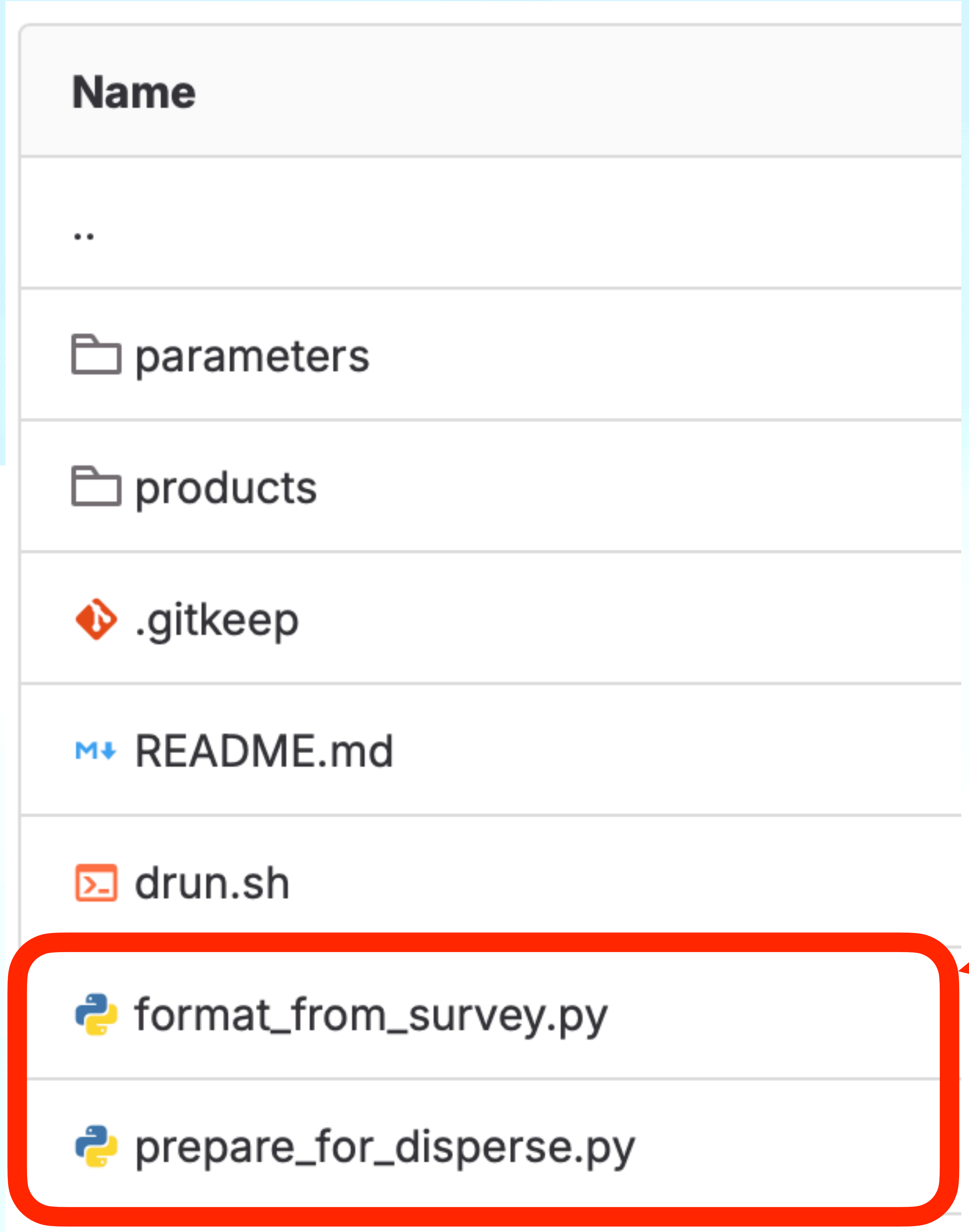


```
# activate the virtual env
. ~/env/bin/activate

# install the requirements
pip install -r ~/step3_analysis_and_plots/setup/requirements.txt
```

Example of install-user-deps.sh for a python environment

Run

Analysis scripts



Name
..
parameters
products
.gitkeep
README.md
drun.sh
format_from_survey.py
prepare_for_disperse.py

These are scripts created by the user that perform the analysis.

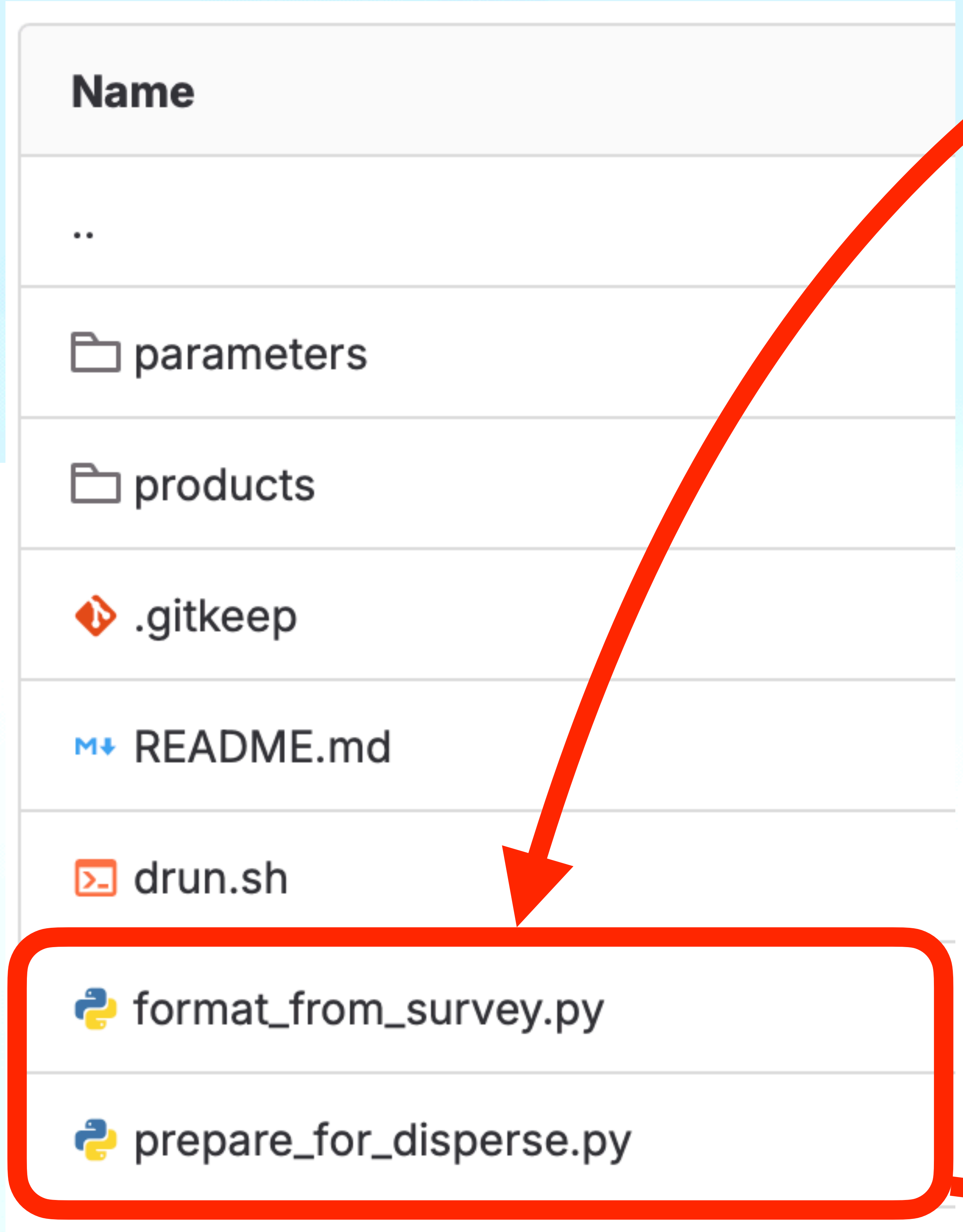
They are executed as they are within the container created by the installation procedure.

Example of analysis scripts for the get_data step

Run

drun.sh

The analysis scripts are called by the drun.sh script which is executed inside the container.



Name

..

parameters

products

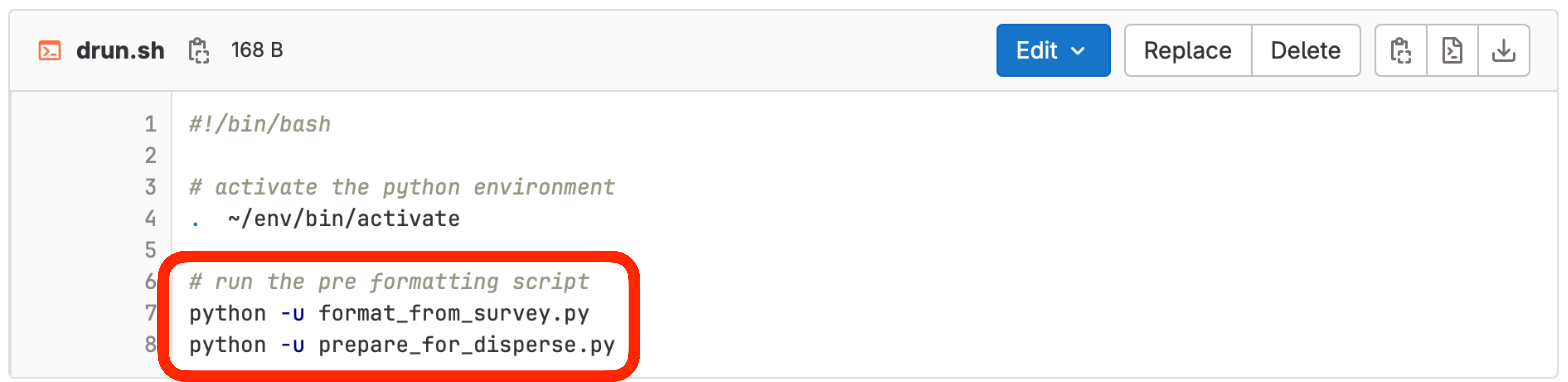
.gitkeep

README.md

drun.sh

format_from_survey.py

prepare_for_disperse.py

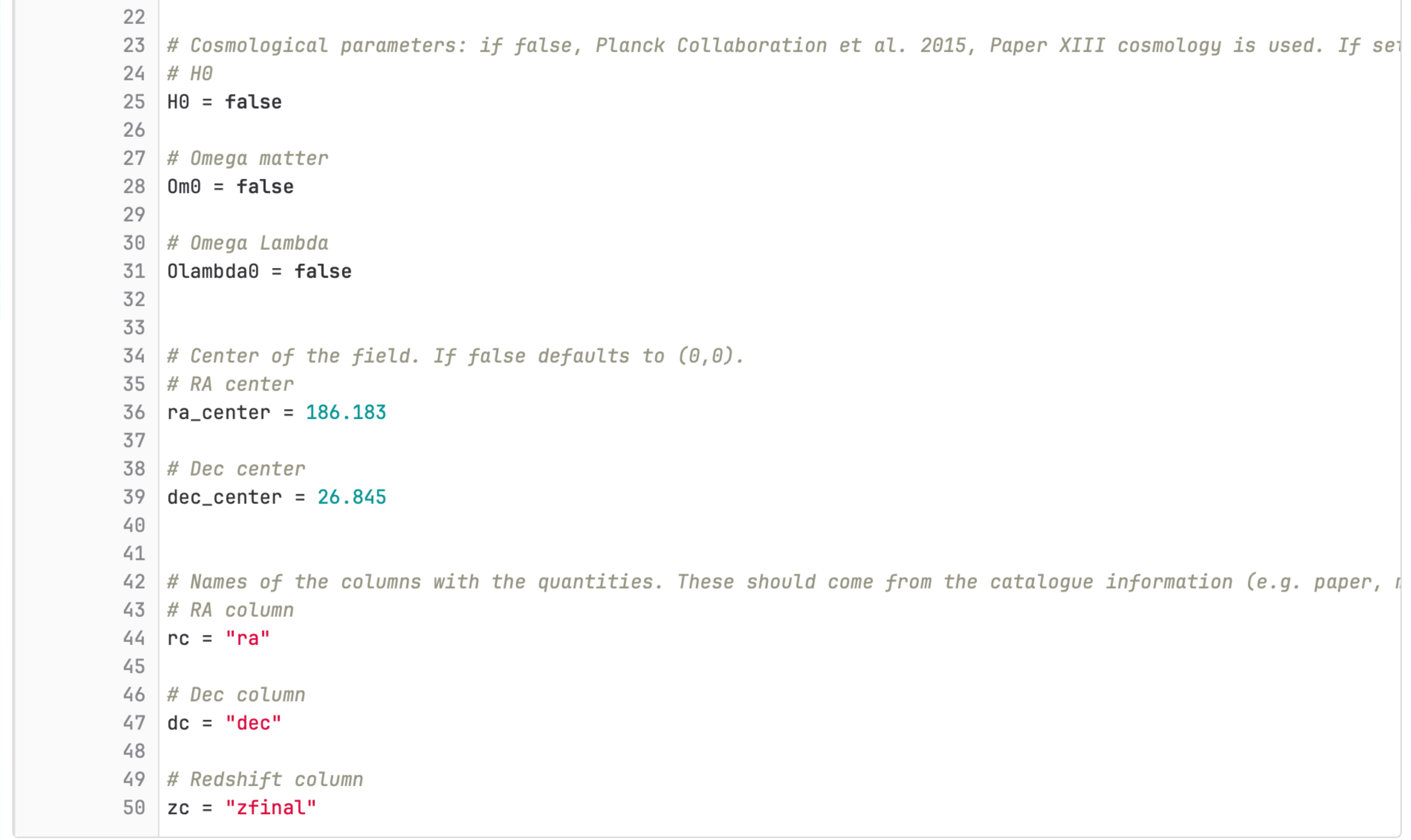


```
#!/bin/bash

# activate the python environment
.  ~/env/bin/activate

# run the pre formatting script
python -u format_from_survey.py
python -u prepare_for_disperse.py
```

Example of analysis scripts for the get_data step

Parameters

Toml file



```

# Cosmological parameters: if false, Planck Collaboration et al. 2015, Paper XIII cosmology is used. If se
# H0
H0 = false

# Omega matter
Om0 = false

# Omega Lambda
Olambda0 = false


# Center of the field. If false defaults to (0,0).
# RA center
ra_center = 186.183

# Dec center
dec_center = 26.845


# Names of the columns with the quantities. These should come from the catalogue information (e.g. paper,
# RA column
rc = "ra"

# Dec column
dc = "dec"

# Redshift column
zc = "zfinal"
```

These are parameters that can be used in an analysis by the analysis routines. They are looked for in the parameters folder.

SciTraceWeb

DRP creation and (re-)use in a user friendly way

What is SciTraceWeb?

SciTraceWeb is an instance of SciTrace running at AIP, accessible through a web page.

It allows efficient and user-friendly DRP creation, as well as possibilities for DRP manipulation such as:

- Exploration: a DRP can be accessed and its content inspected but not modified.
- Run: a DRP is executed.
- Modification: a DRP can be accessed, its content modified, and a new DRP is created, the difference between the two is recorded by the system.

Creation

- DRP creation starts with the creation of a GitLab repository.
- The structure of the repository is fixed, with the necessary files found at the correct position.
- A given DRP can be based on a previous one. In that case it will have access to the previous's code and environment.
- Input data can be mounted in the data folder. They can be the products of a previous step.
- The scripts install.sh, intall-deps.sh, and install-user-deps.sh are run automatically.

DRP Run

- Right after creation a DRP has no products. To create the products it must be run.
- Running happens within the container.
- The script drun.sh is executed automatically.
- It then runs the analysis scripts.
- Running the container uses resources.

Exploration

- Package exploration is possible thanks to a JupyterLab instance installed within the container.
- Data are accessible, so are products, and the analysis code.
- Exploration is performed in read-only mode.

Modification

- Modification is similar to exploration but it allows also writing.
- Several operations are possible: the code can be modified and run, data accessed, parameters modified.
- GitLab integration means that once the modifications are done they can be saved, pushed to a cloned version of the repository and a new package created and run.
- Parameters can also be downloaded, modified, and re-uploaded to generate a new DRP.

Tracing

- All the operations described before are traced via the hash of the container image.

- Package creation generates an hash.

- A new hash is generated for package run that indicates that products have been created.

- Package modification also generates a new hash, different from the starting one to indicate that the package has been modified and is different from the original.

- There is a DRP registry where generated DRPs are saved and can be explored/used as starting point by the community.

Practical example

The analysis of the cosmic web around the Coma cluster as detected by DisPerSE implemented in SciTraceWeb

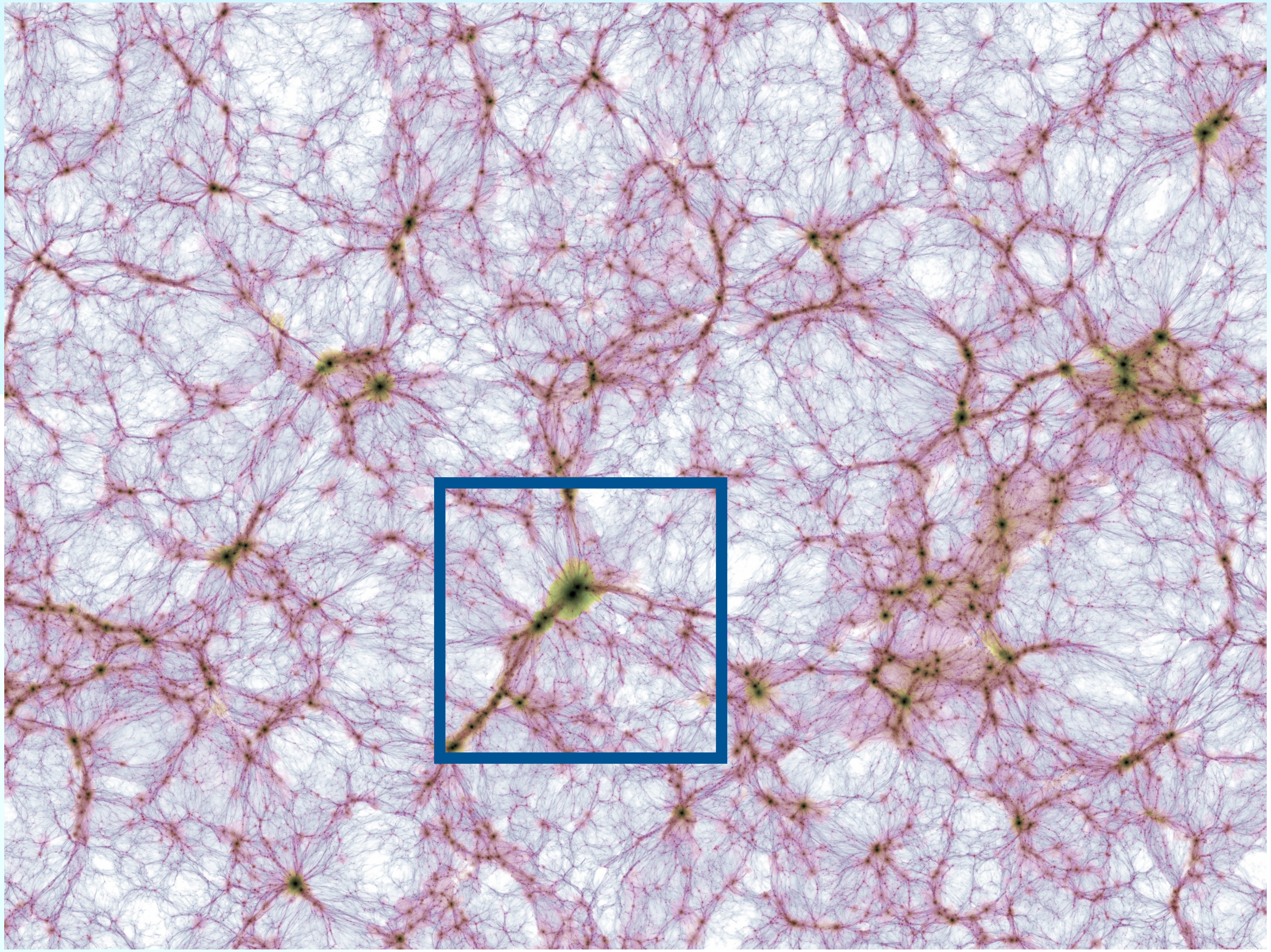
Scientific analysis based on:
Malavasi et al. 2020a, A&A, 634, A30
Malavasi et al. 2020b, A&A, 642, A19
Malavasi et al. 2023, A&A, 675, A76

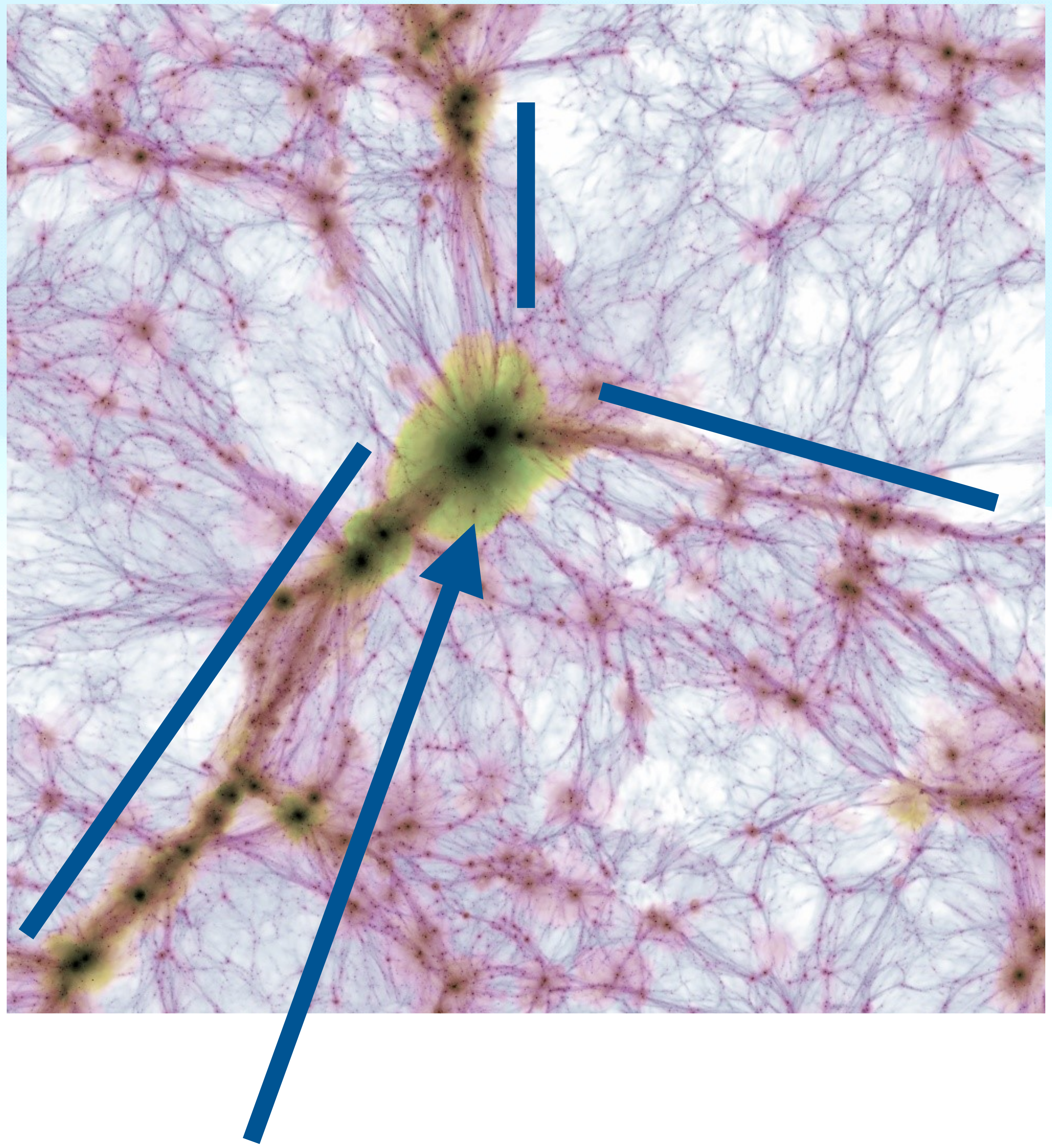
Implementation in PUNCH based on:
Fournier et al., in prep.
Malavasi et al., in prep.

The cosmic web

Network of connected structures made of galaxies, dark matter, and gas.

The cosmic web impacts the formation of structures and the formation of galaxies.





Clusters as nodes connected by filaments.

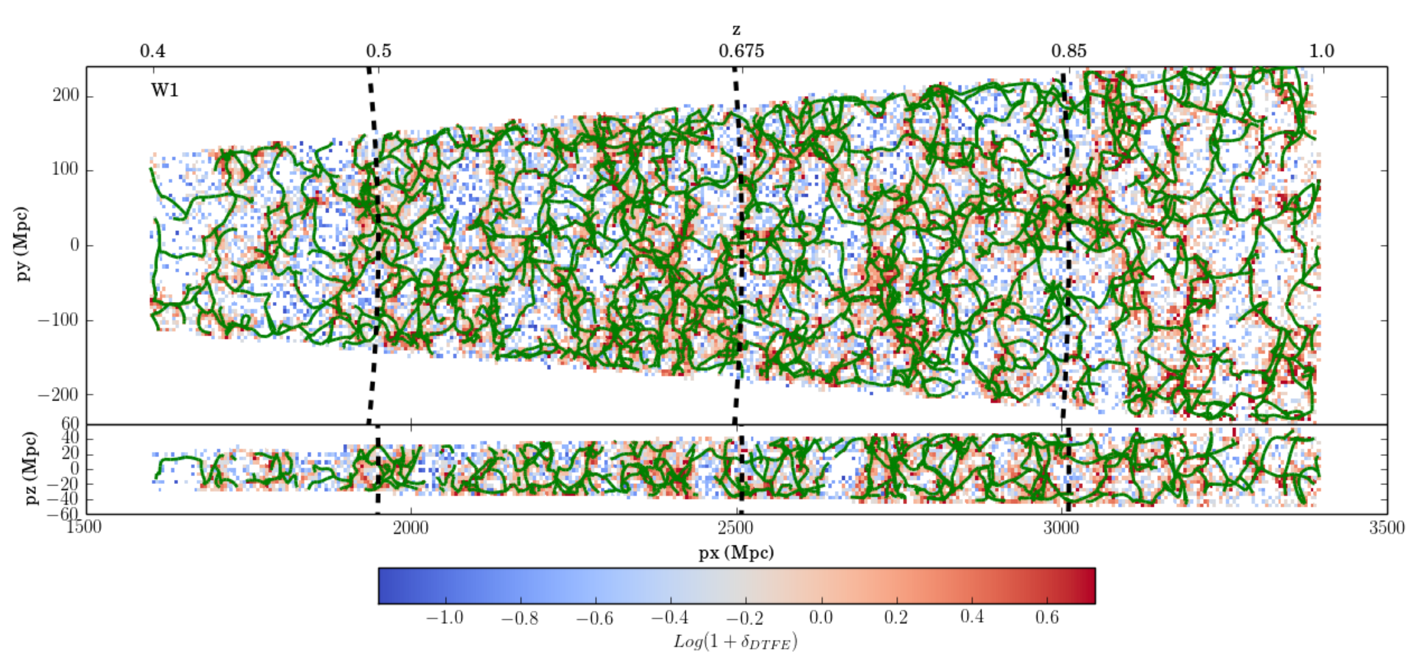
Credits: IllustrisTNG

Aragón-Calvo+10, Cautun+14, …

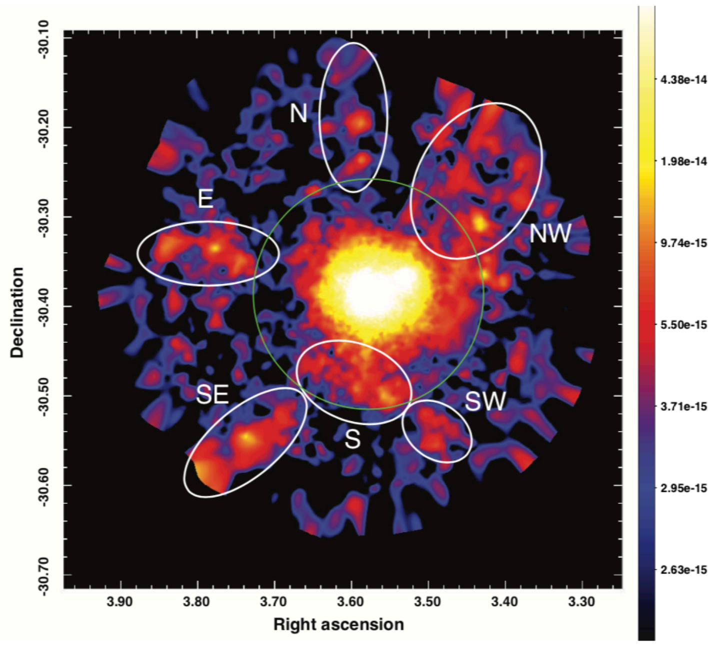
Our goal

- Study the connections and the connectivity of a massive, nearby, well known cluster.
- Study the LSS in a large region around a cluster.
- Perform a case study of a well known object.
- Study filament properties.

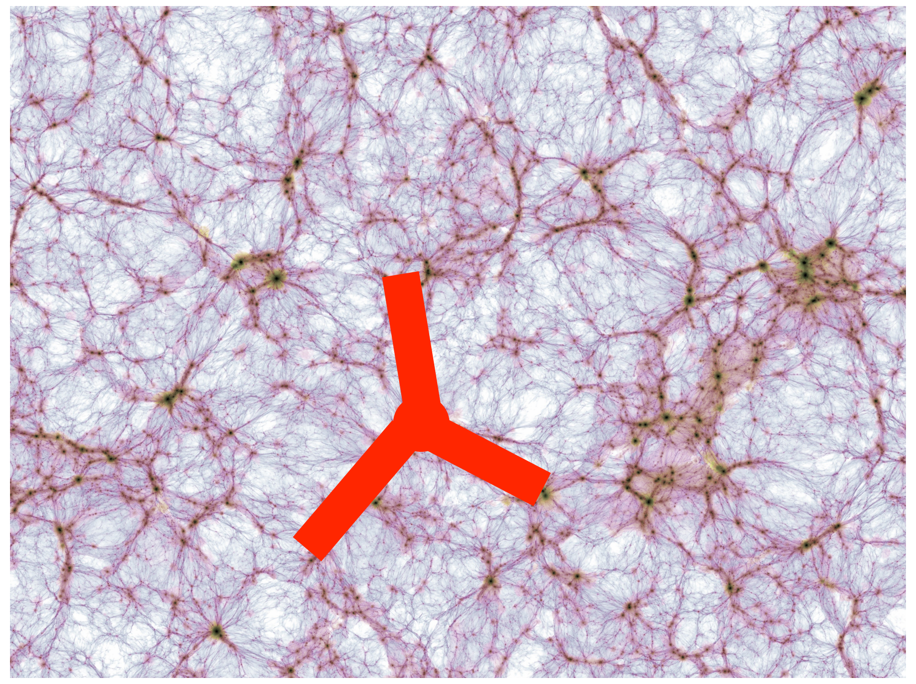
Apply a filament detection algorithm to a spectroscopic survey, then study the properties of the filaments detected around a cluster.



+



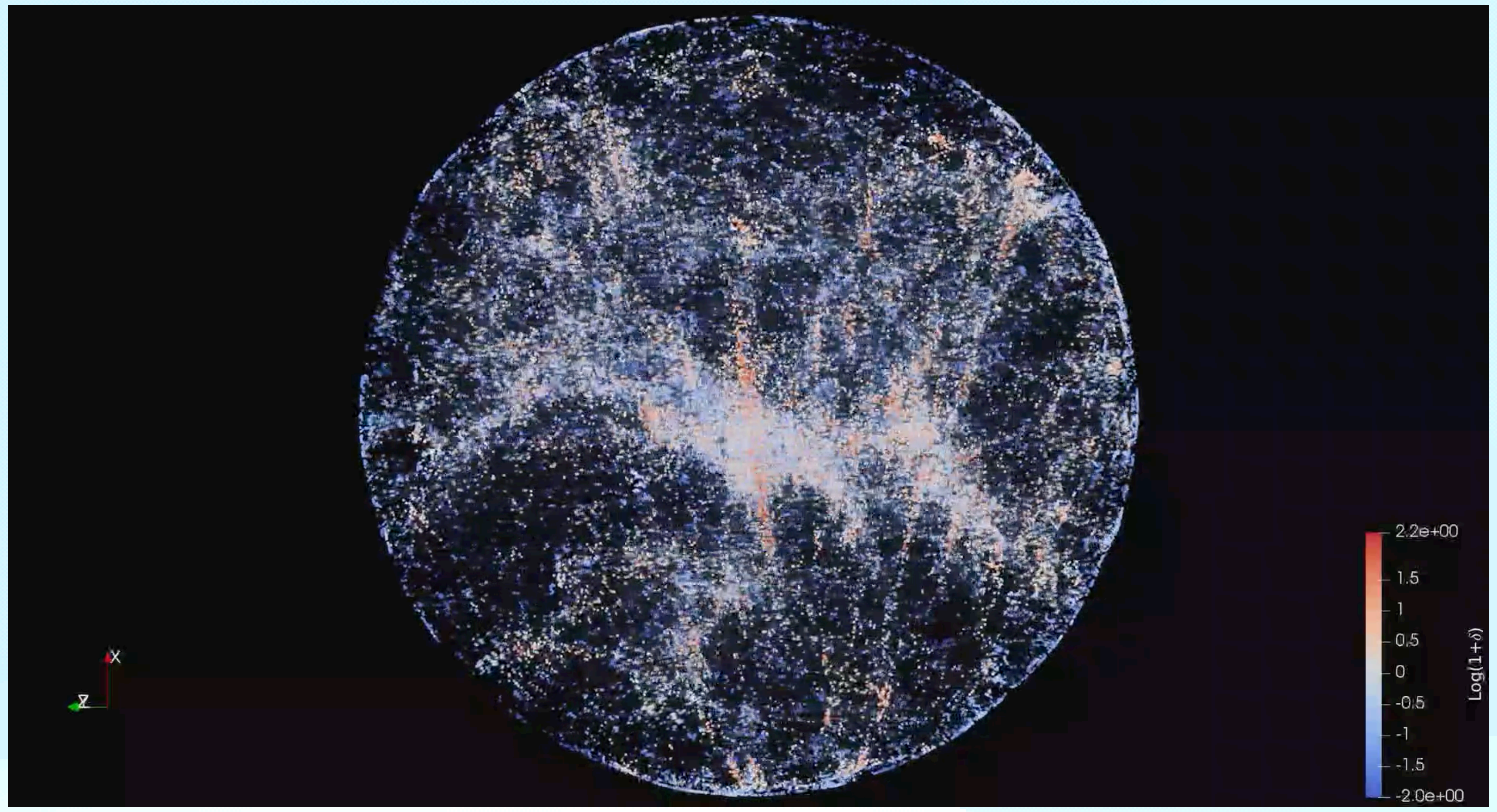
=

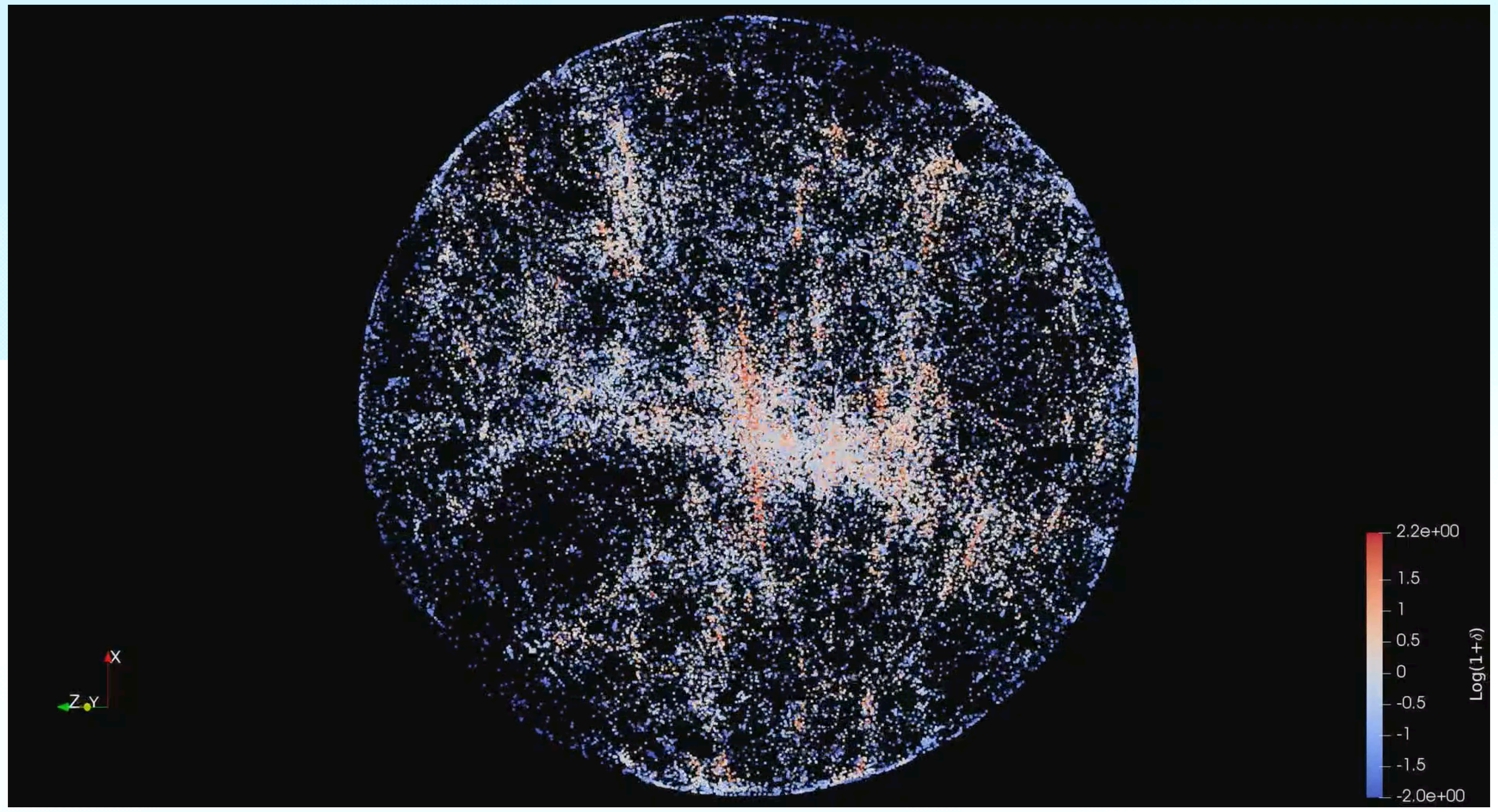


Discrete Persistent Structure Extractor

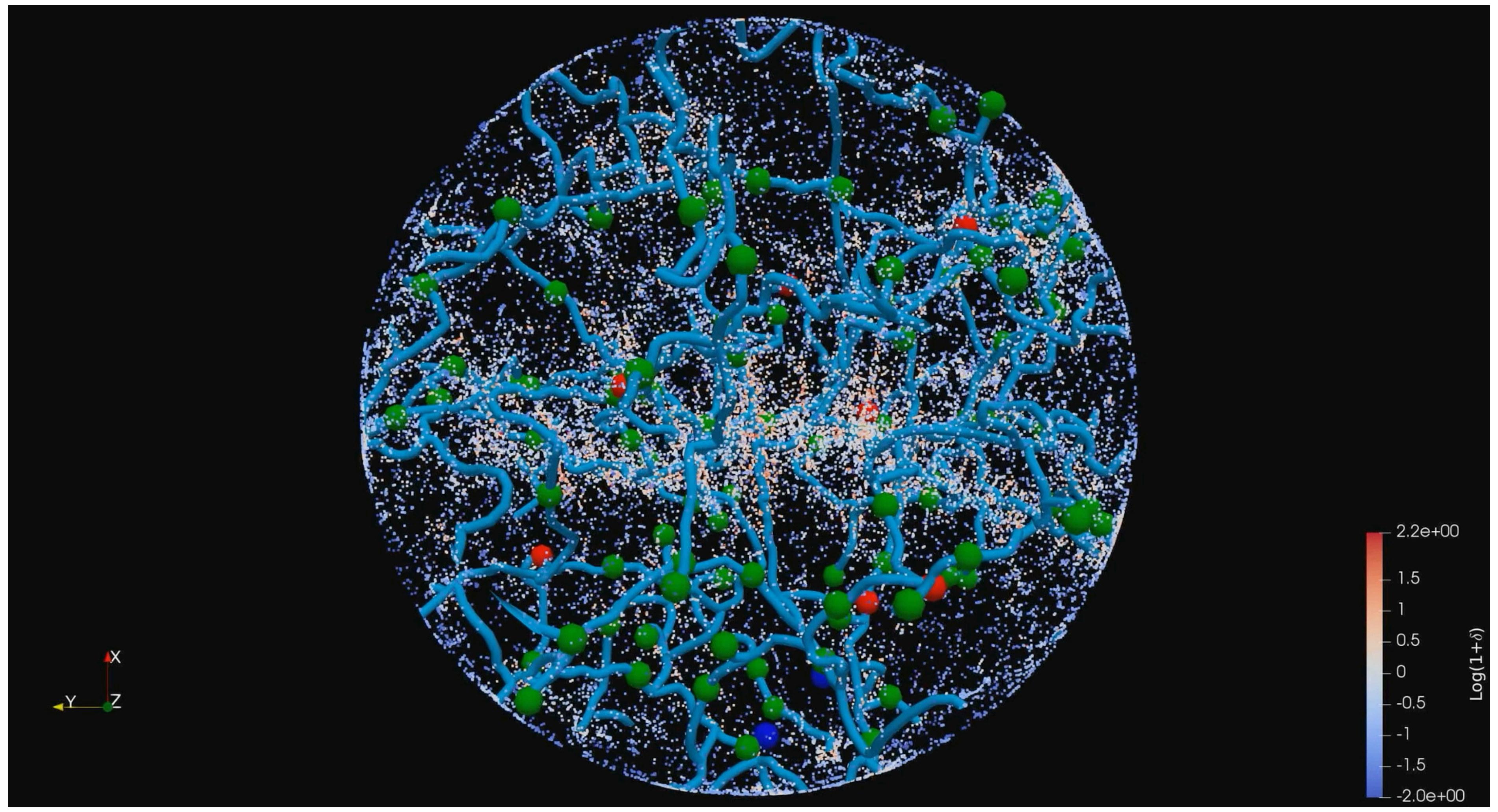
Sousbie 2011, Sousbie et al. 2011

Measure of the density field (DTFE).
Possibility of smoothing (although not necessary).





Computation of the discrete gradient.
Detection of the critical points (maxima, minima, and saddles).

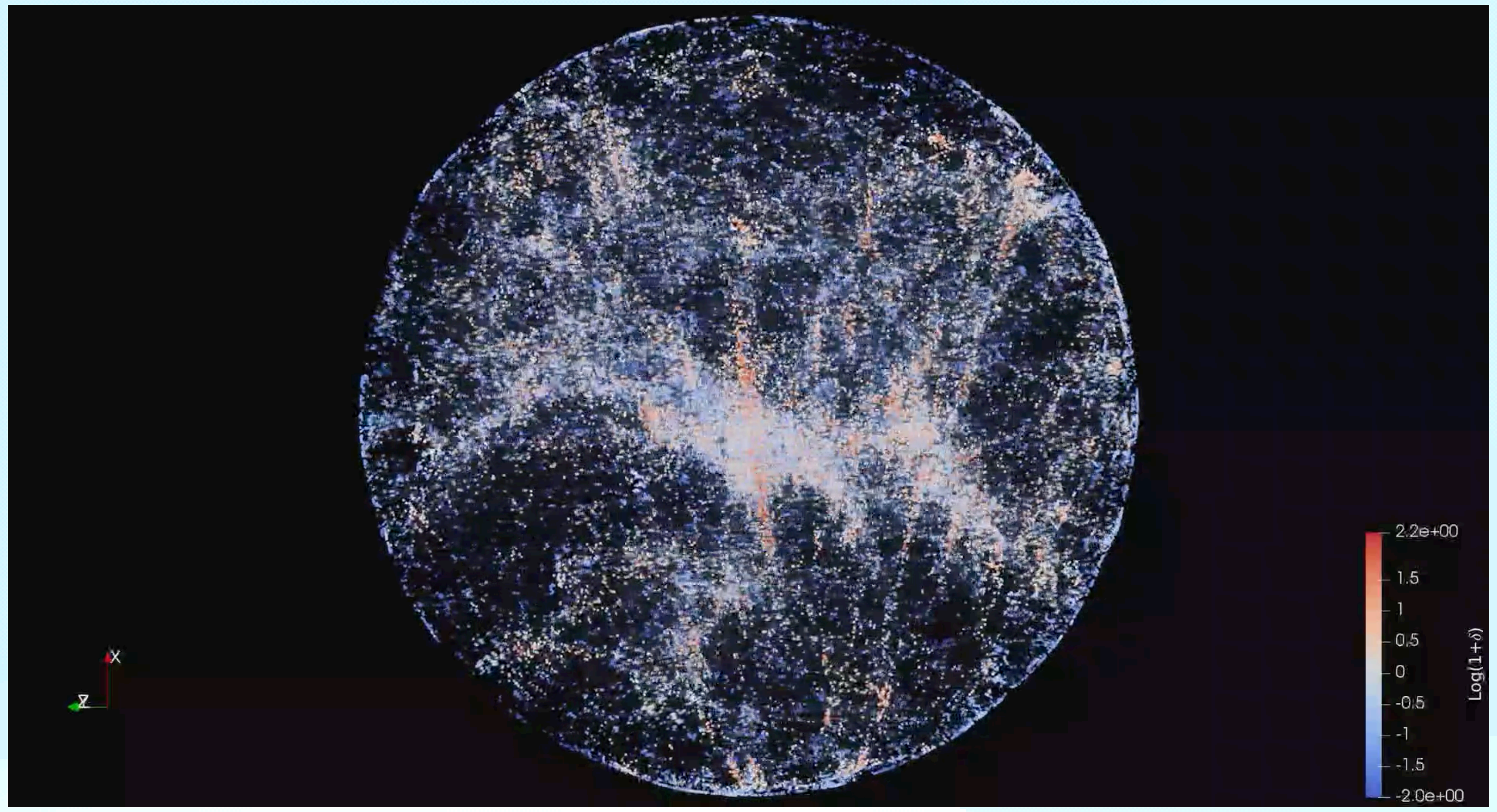


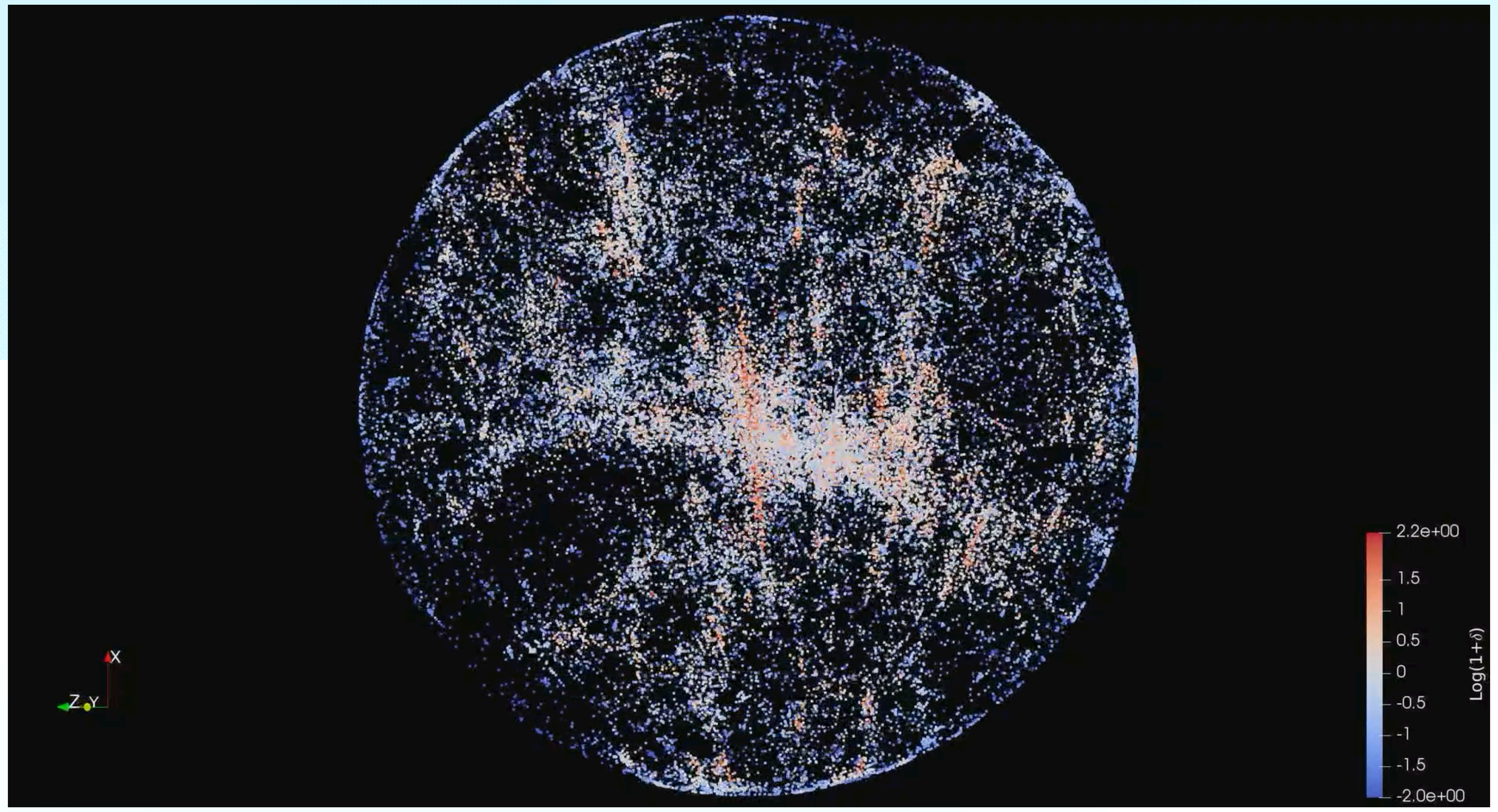
Connection of the critical points (maxima and saddles) with filaments. Persistence cut to eliminate spurious structures due to noise (expressed in terms of numbers of sigma, similar to S/N threshold).

Discrete Persistent Structure Extractor

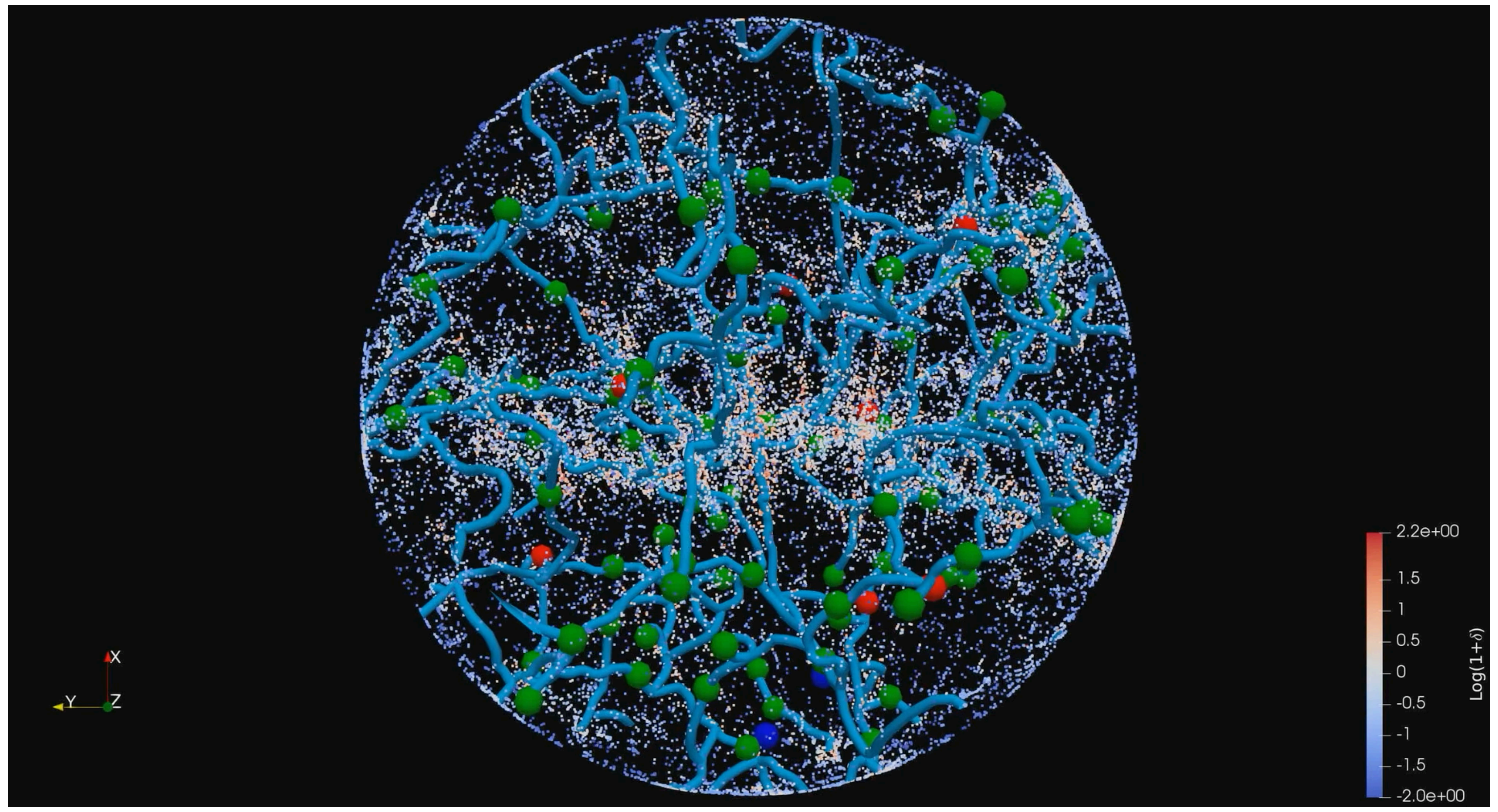
Sousbie 2011, Sousbie et al. 2011

Measure of the density field (DTFE).
Possibility of smoothing (although not necessary).





Computation of the discrete gradient.
Detection of the critical points (maxima, minima, and saddles).

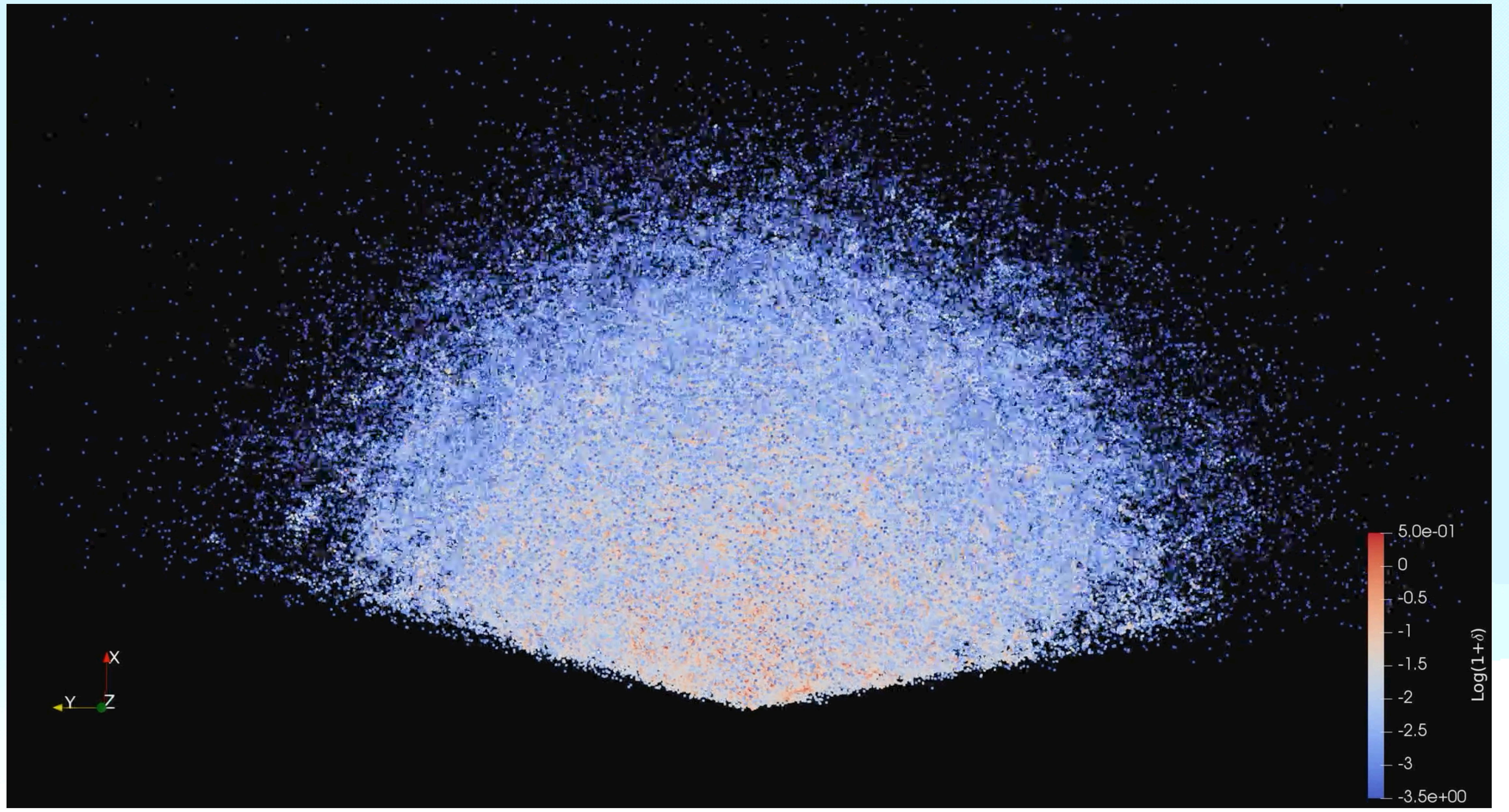


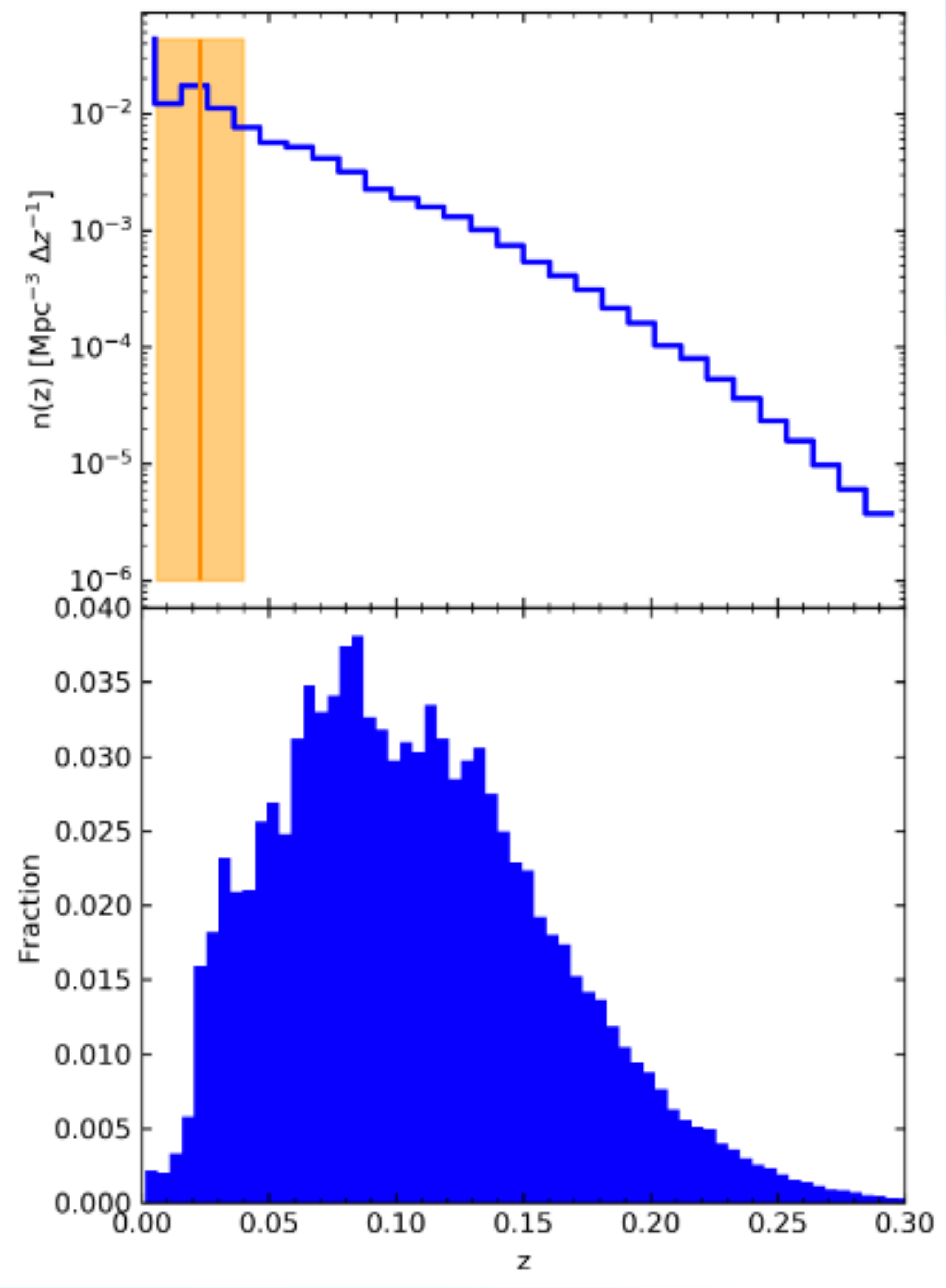
Connection of the critical points (maxima and saddles) with filaments. Persistence cut to eliminate spurious structures due to noise (expressed in terms of numbers of sigma, similar to S/N threshold).

Tracers of the filaments: SDSS

SDSS DR7 Legacy survey Main Galaxy Sample (Strauss+02) ~600 000 galaxies at z = 0-0.3.

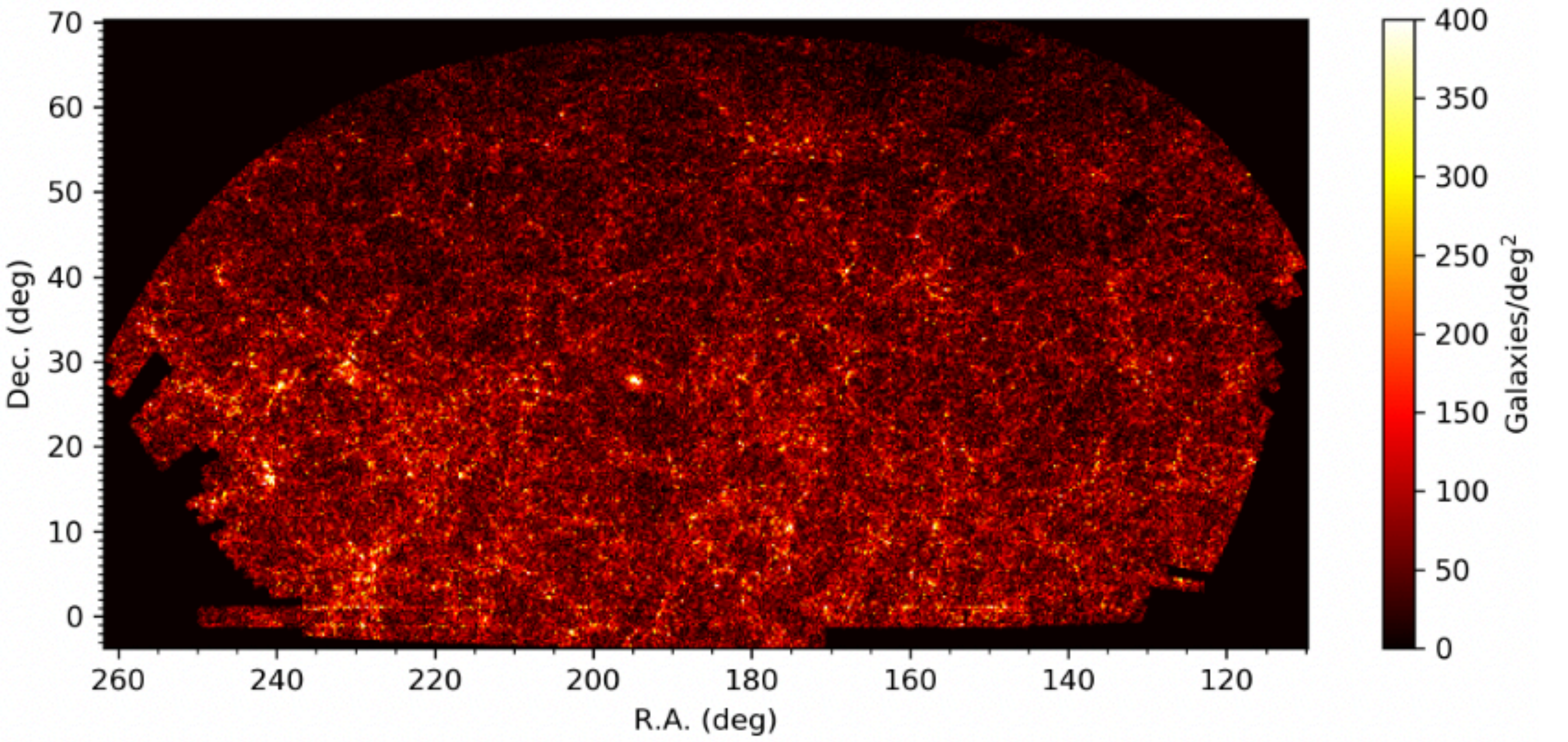
Uniform coverage in the plane of the sky and smooth redshift distribution.





Volume density distribution: n(z)

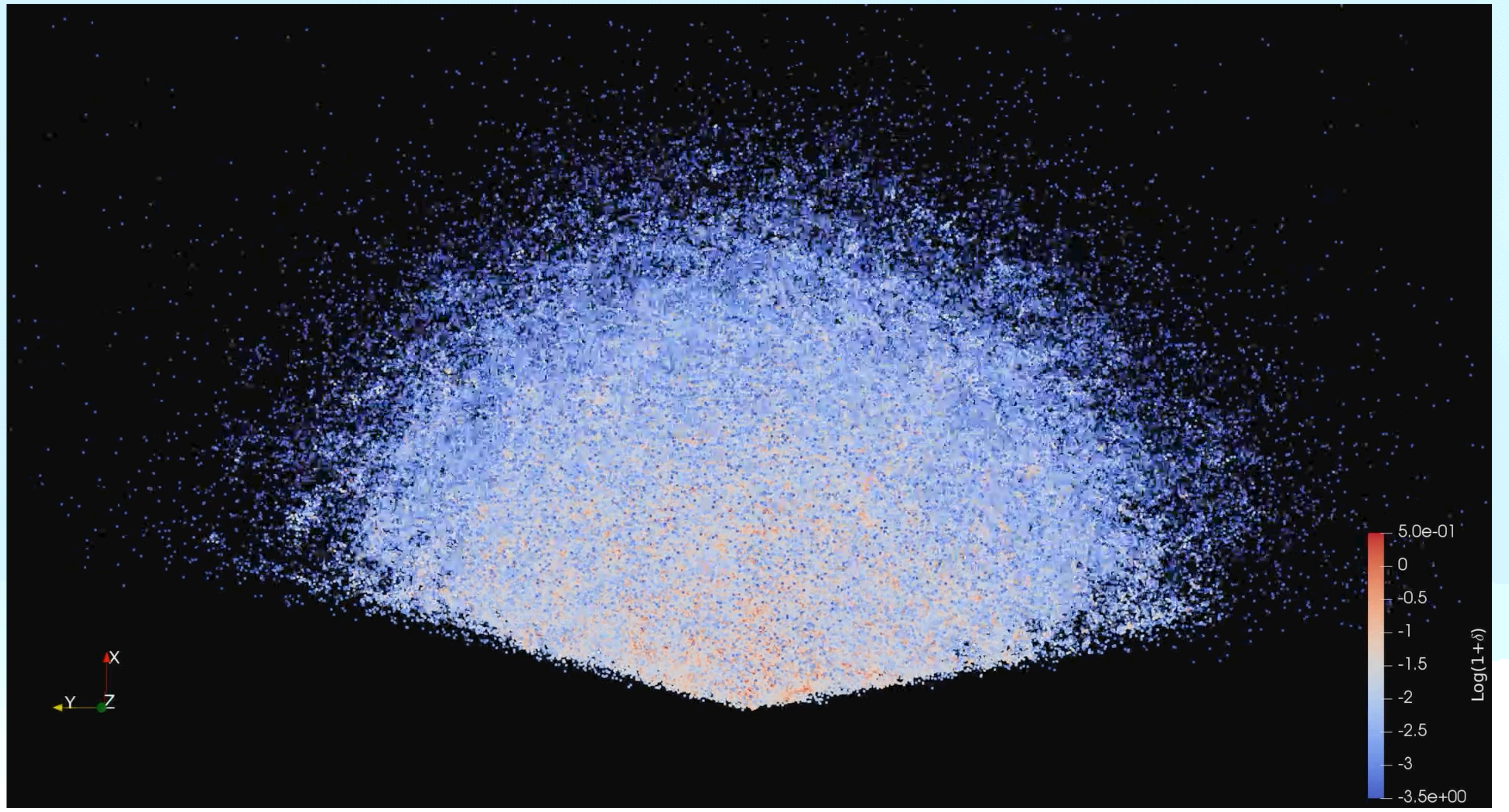
Redshift distribution

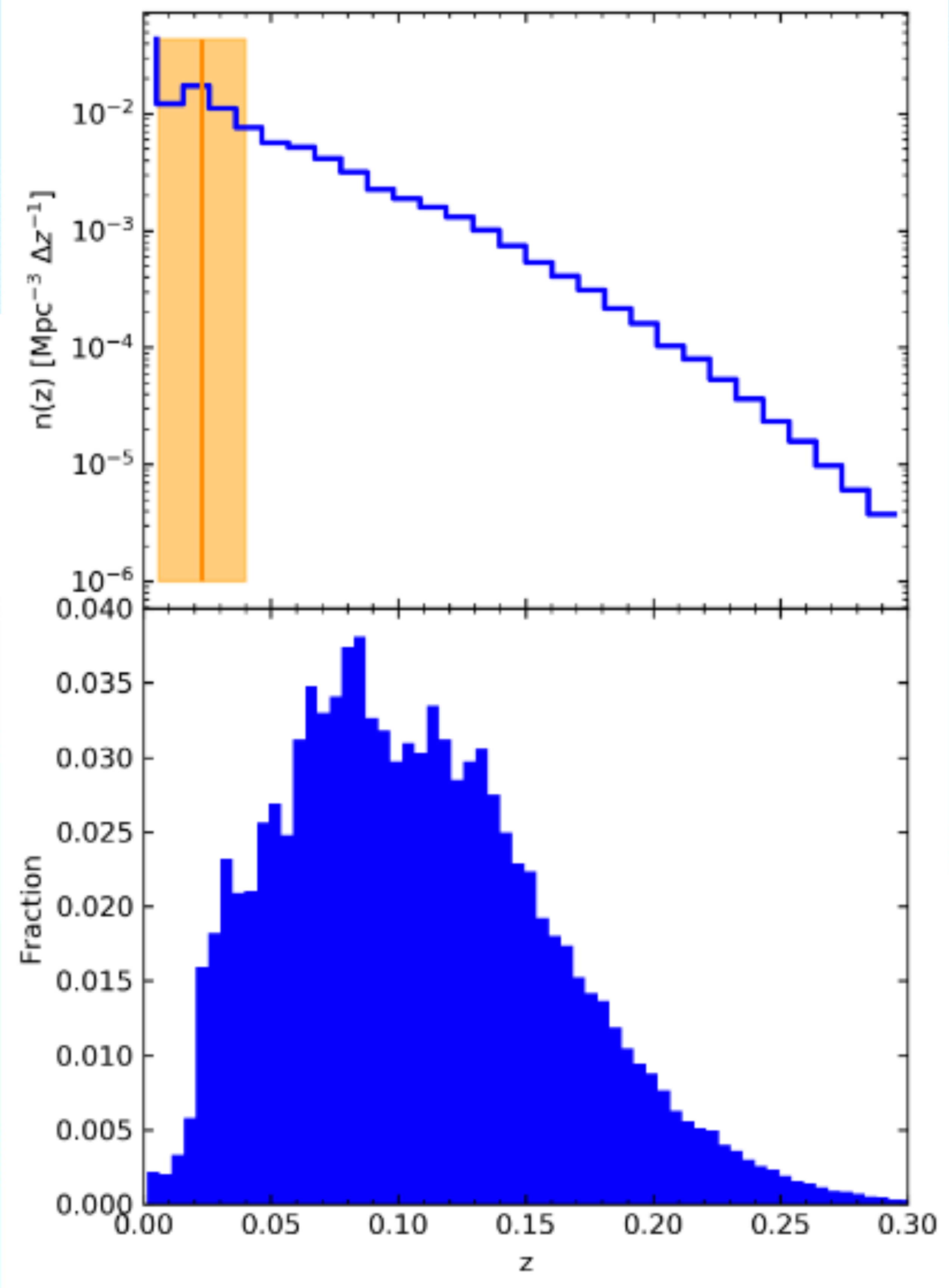


Tracers of the filaments: SDSS

SDSS DR7 Legacy survey Main Galaxy Sample (Strauss+02) ~600 000 galaxies at z = 0-0.3.

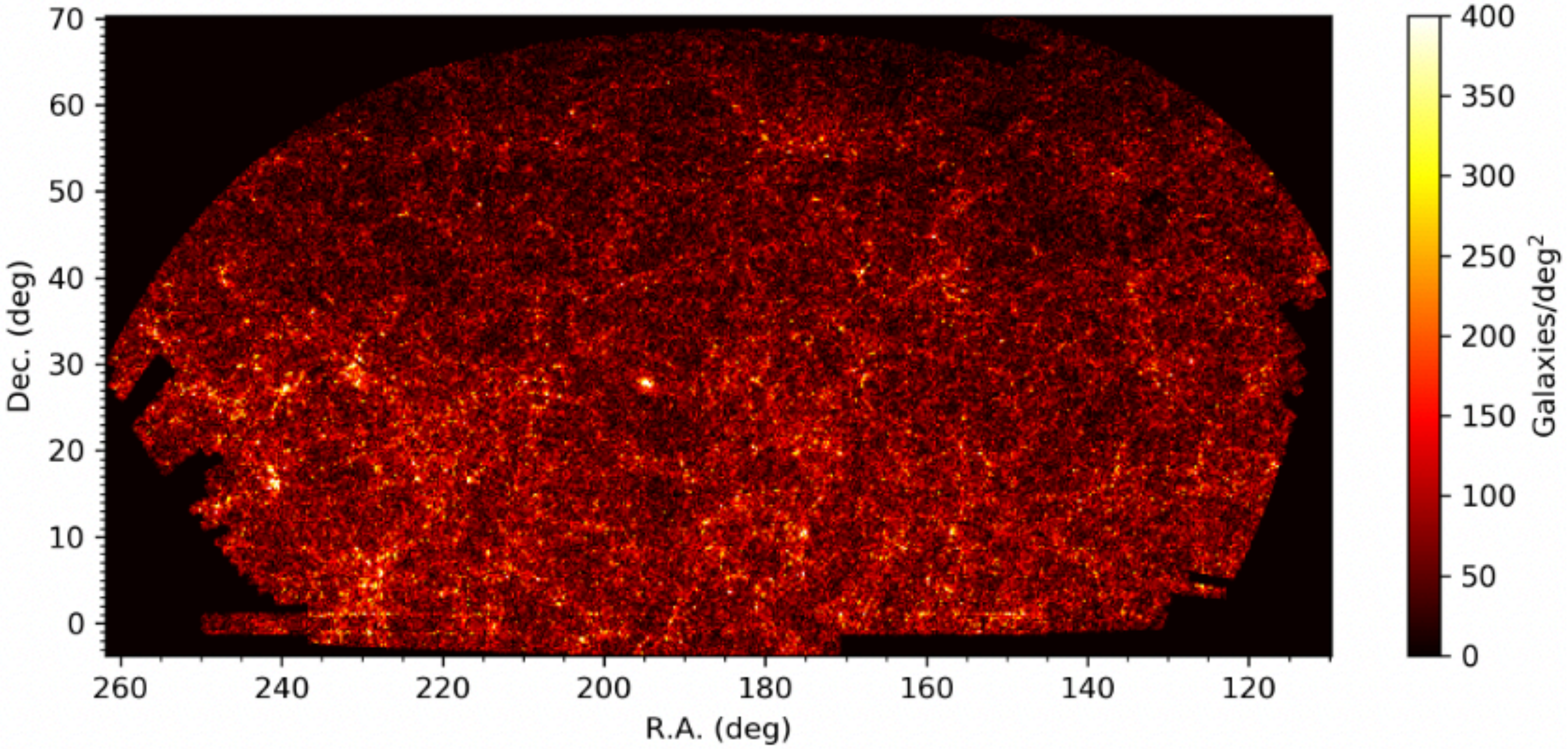
Uniform coverage in the plane of the sky and smooth redshift distribution.





Volume density distribution: n(z)

Redshift distribution

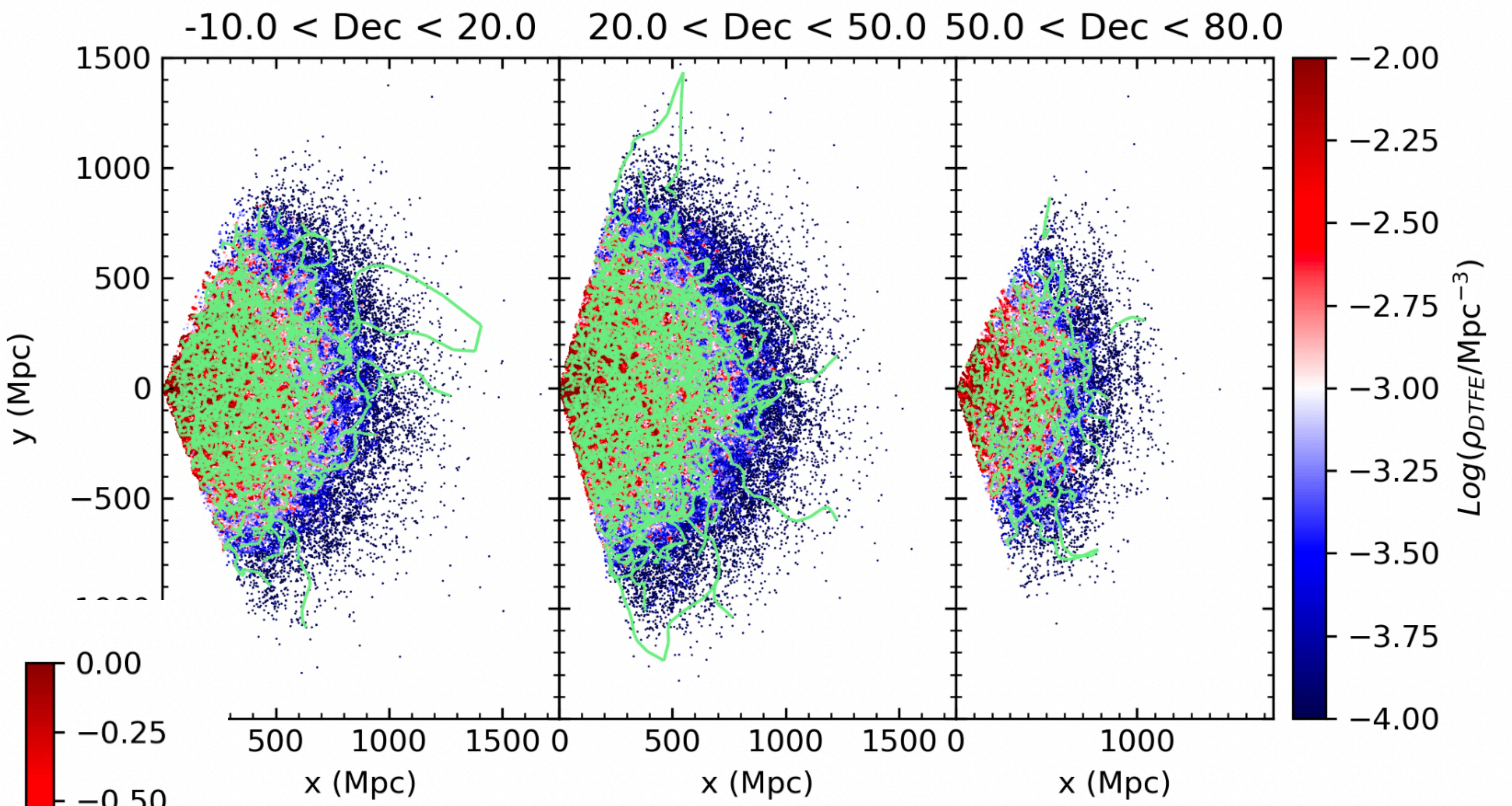


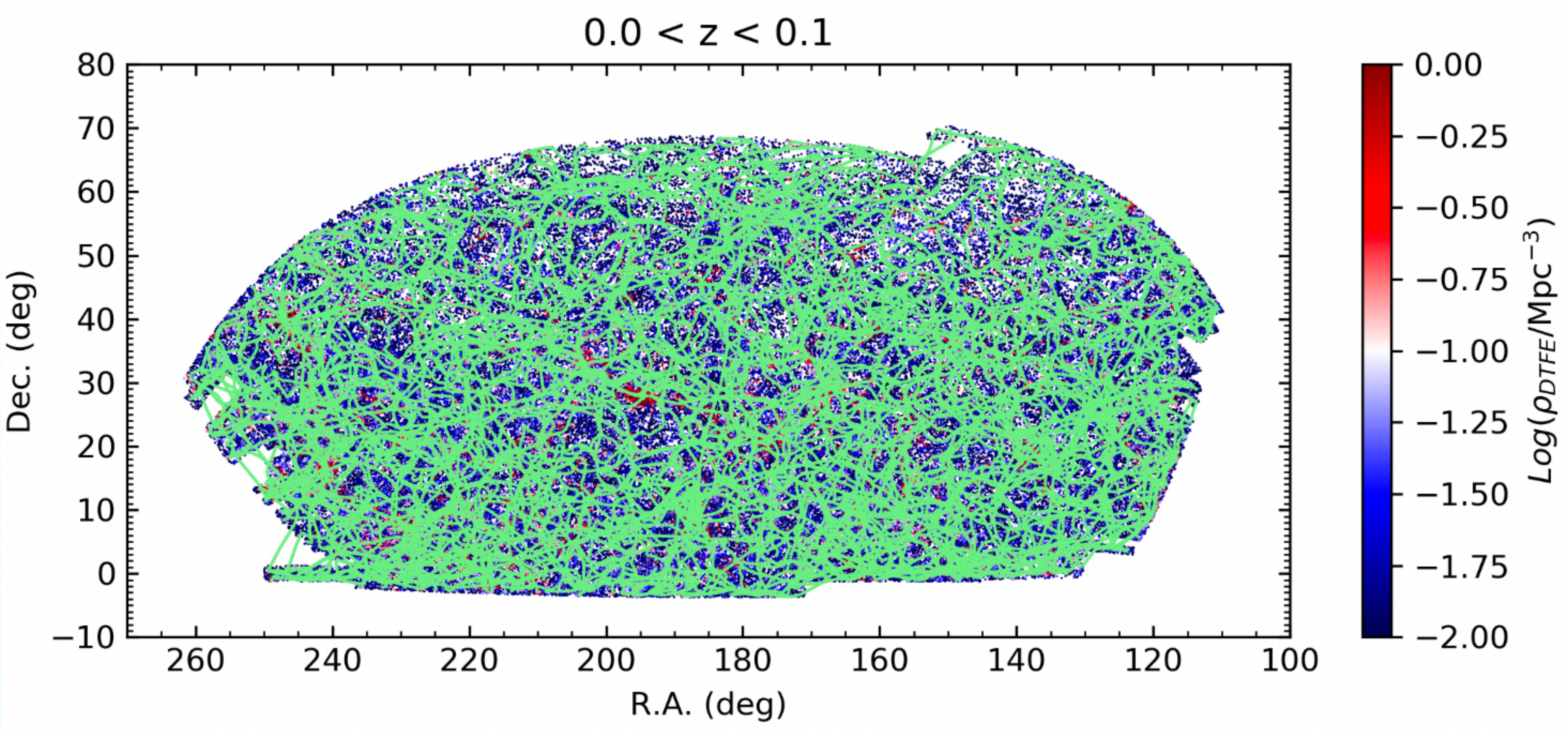
Catalogue of filaments in the SDSS

Malavasi et al. 2020b

We released the catalogue of filaments for use by the community.

Full characterization of properties and systematics.



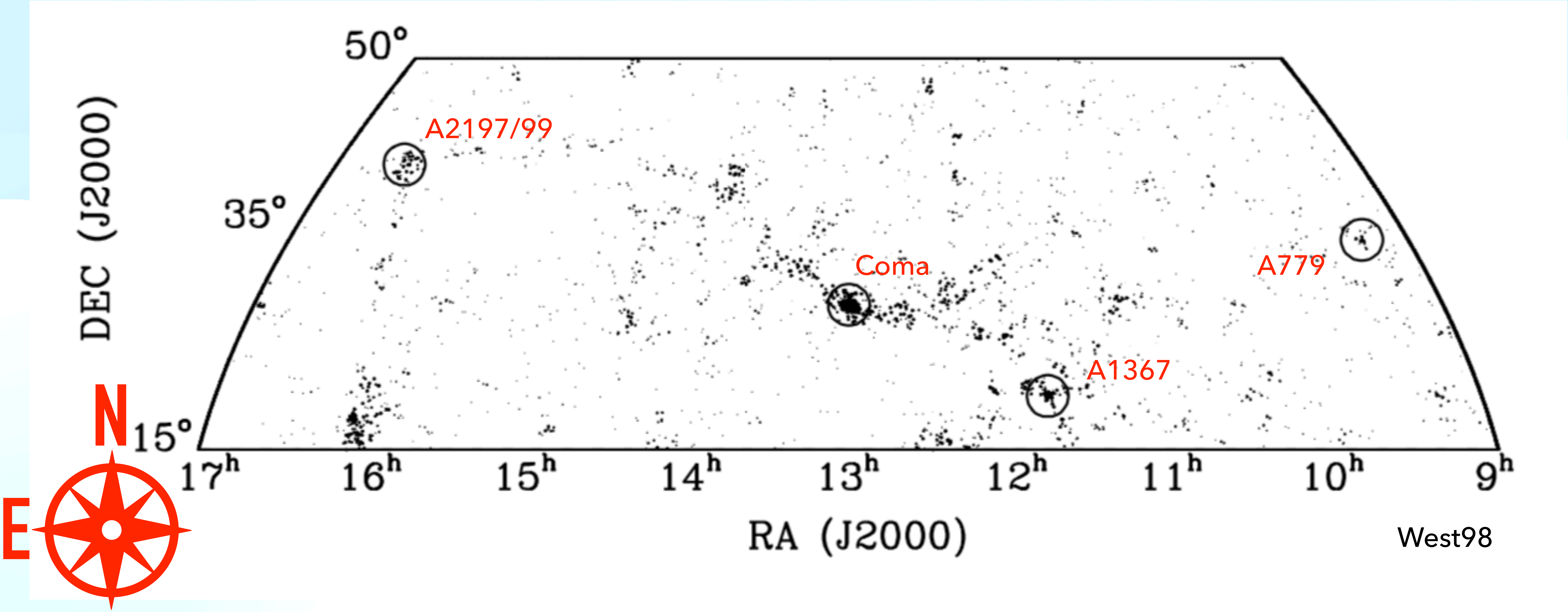


Available for DR7 and DR12.

https://l3s.osups.universite-paris-saclay.fr/cosfil.html

Coma and its LSS

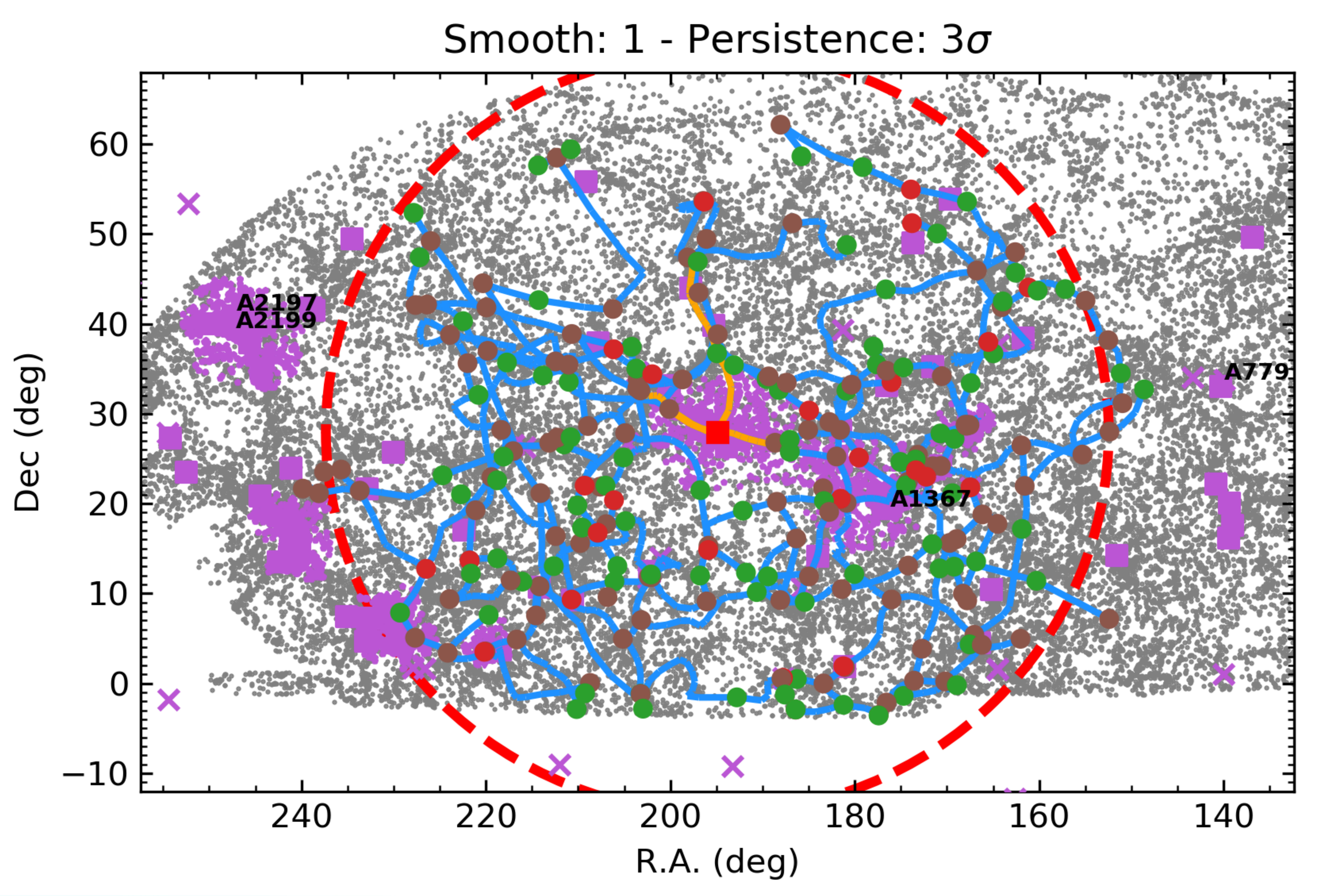
- Physical parameters (Mass, radius, velocity dispersion, Łokas&Mamon03)
- Substructure (Subgroups, ICL) (Adami+05a, 05b, 09)
- Idea of the LSS in the region (other clusters in the region)



West98

Filaments around the Coma cluster

DisPerSE: Sousbie11, Sousbie+11 - SDSS DR7: Abazajian+09, Strauss+02



- Discrete Persistent Structure Extractor (DisPerSE) applied to Sloan Digital Sky Survey DR7.

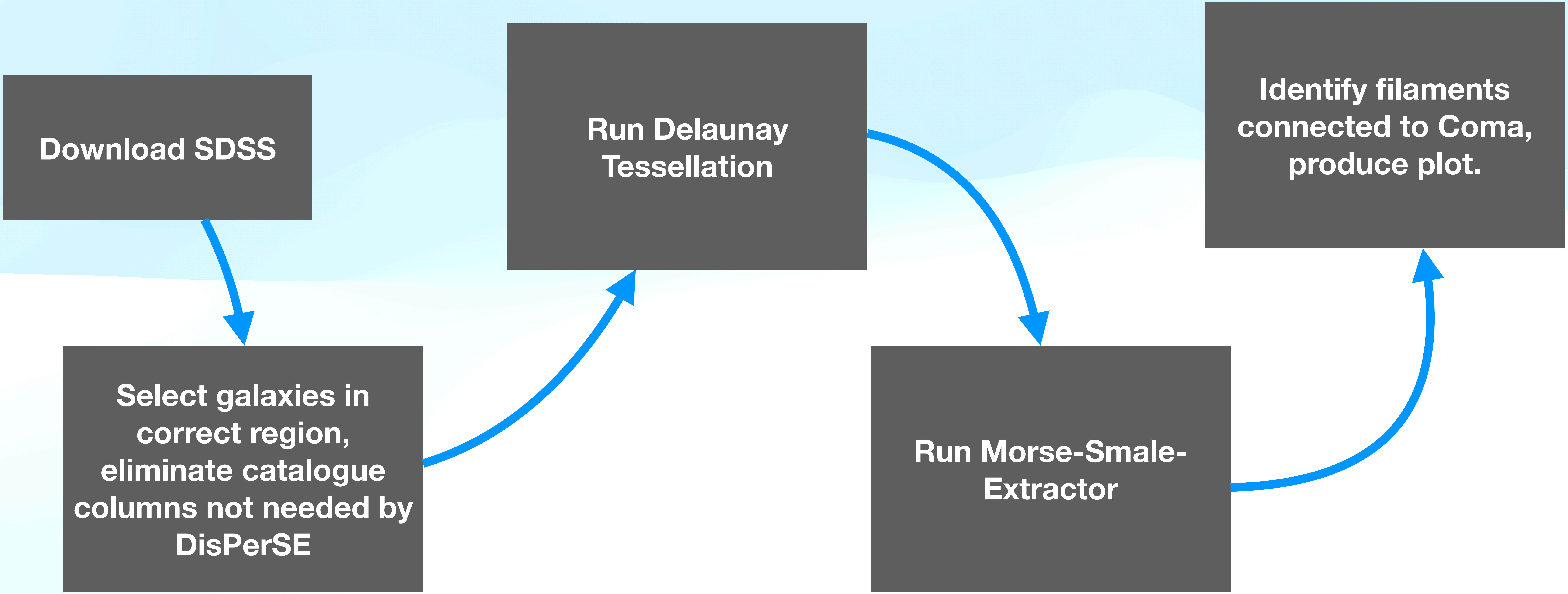
- Focus on the Coma cluster: well known cluster with a lot of information available.

- Detected three filaments connected to the Coma cluster.

Implementing the analysis in SciTraceWeb

- Breakdown the analysis in sequential steps.
- Identify input and output.
- Identify parameters.
- Write script to install the code and its dependencies.
- Optimize and tidy up the code: your package will be explored by others.
- Create repository with the correct structure.
- Login to SciTraceWeb app and implement.

Identifying steps in the analysis



Download SDSS

Select galaxies in correct region, eliminate catalogue columns not needed by DisPerSE

Run Delaunay Tessellation

Run Morse-Smale-Extractor

Identify filaments connected to Coma, produce plot.

Identifying steps in the analysis

1
Download SDSS
Select galaxies in correct region, eliminate catalogue columns not needed by DisPerSE
2
Run Delaunay Tessellation
3
Run Morse-Smale-Extractor
4
Identify filaments connected to Coma, produce plot.

Input & output

- Step 1: no input, output is a catalogue of galaxies ready for DisPerSE
- Step 2: input is catalogue output by step 1, output is tessellation
- Step 3: input is tessellation, output is skeleton
- Step 4: input is skeleton AND catalogue output by step 1, output is plot (PNG)

Parameters

- Step 1: file names, whether to output intermediate catalogues, coordinate centers and cosmology for coordinate conversion
- Step 2: DisPerSE parameter, number of smoothing cycles
- Step 3: DisPerSE parameter, persistence threshold
- Step4: radius up to which search for filaments connected to Coma