

Improvement of predictive Quality for Final Repository Site Simulations through Optimal Data Acquisition and Smart Monitoring

Qian Chen, M.Sc

Prof. Julia Kowalski
Dr. Marc S. Boxberg



Chair of Methods for Model-based Development in
Computational Engineering
Faculty of Mechanical Engineering

Nino Menzel, M.Sc

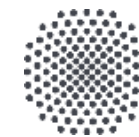
Prof. Florian Wagner



Department of Geophysical Imaging and Monitoring

Maria Fernanda Morales, M.Sc

apl. Prof. Sergey Oladyshkin
Prof. Wolfgang Nowak

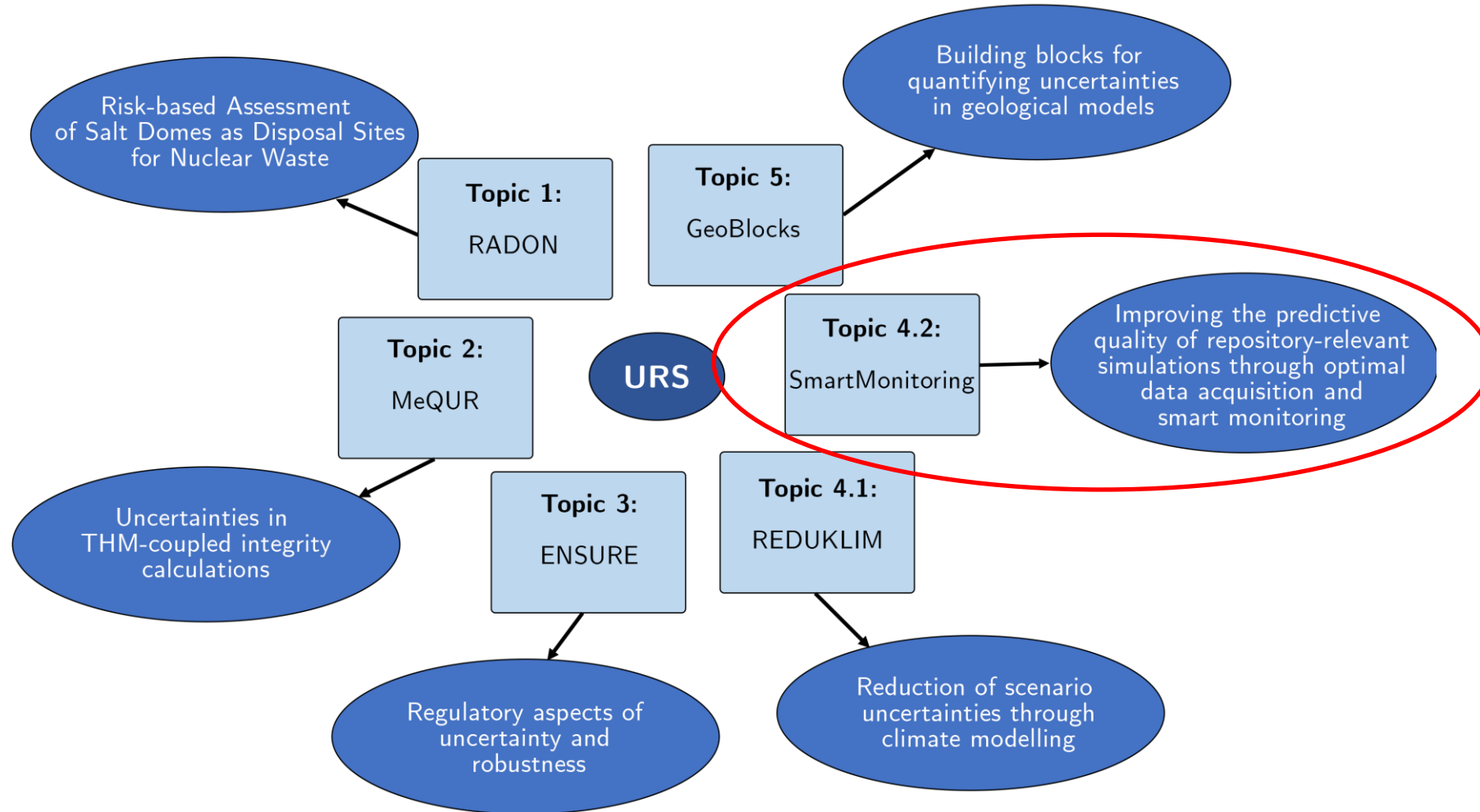


University of Stuttgart



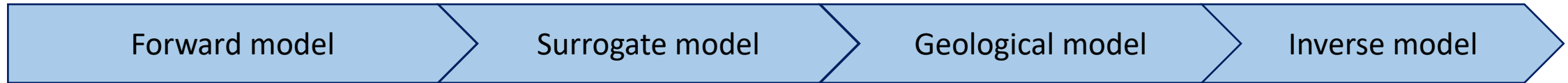
Department of Stochastic Simulation and Safety Research for Hydrosystems
Institute for Modelling Hydraulic and Environmental Systems
Stuttgart Center for Simulation Science

URS Project



Which **type** of field measurements provide the greatest information (reduce uncertainty), and **where** and **when** should these measurements be acquired?

Project workflow



- Create radionuclide transport model (process model).
- The underlying mathematical process model will be extended with regard to selected impact models (the accumulated dose).

Meta-modelling strategy and comparative analysis

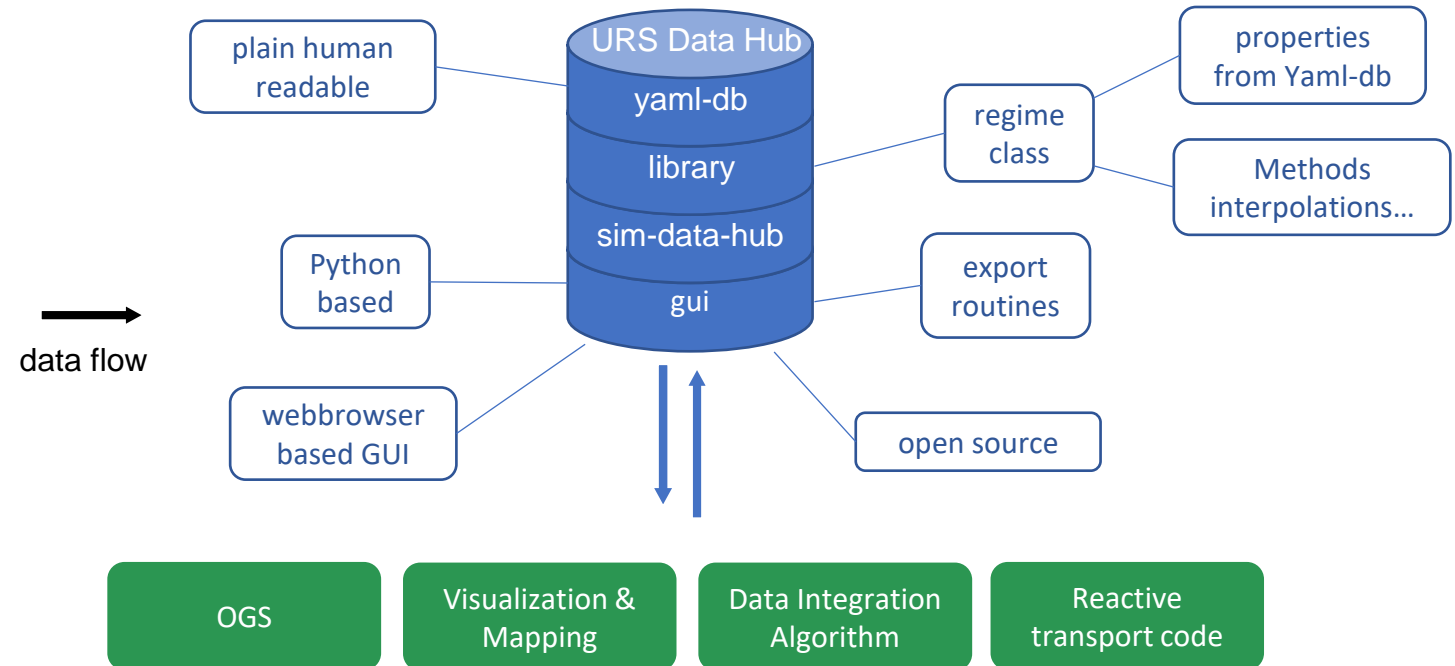
Development of a modern and robust parameter estimation and data assimilation based on metamodels and Bayesian active learning

Transferring the developed methods to real geological models

Simulation-ready benchmark scenarios

We create the reference scenarios including the geological reference models for each possible host rocks of a nuclear waste disposal site.

- One geological model will be selected for each reference rock type.
- The material data will be assembled for each benchmark data pool.
- Parametrization of impact scenarios.
- We follow FAIR research data management principles and realize open access.



URS-data-hub: Data Exchange Interface

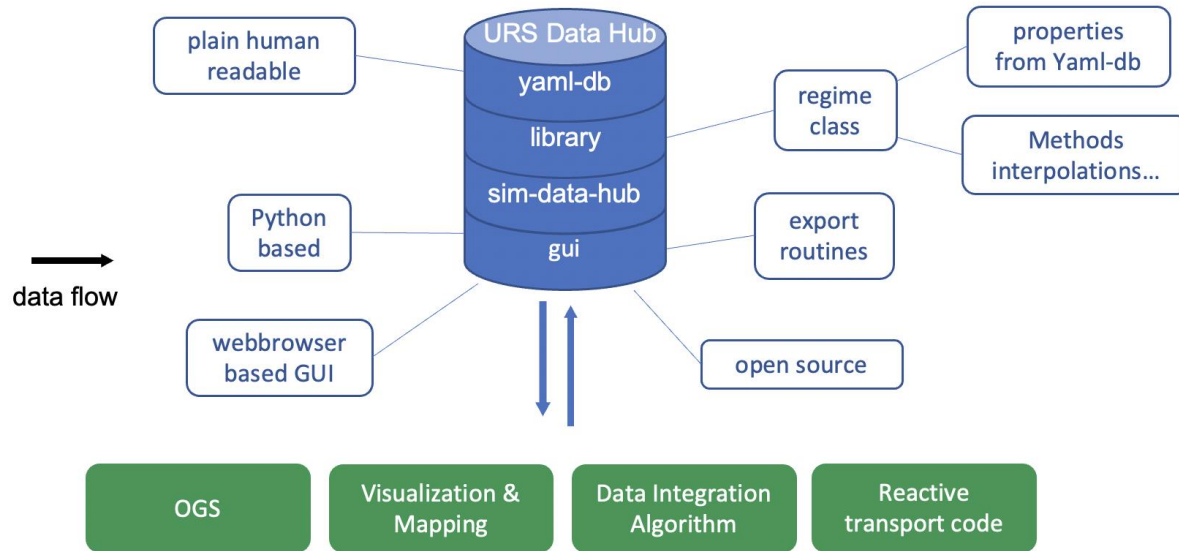
2nd URS PhD Workshop

Date: 17.03.2023

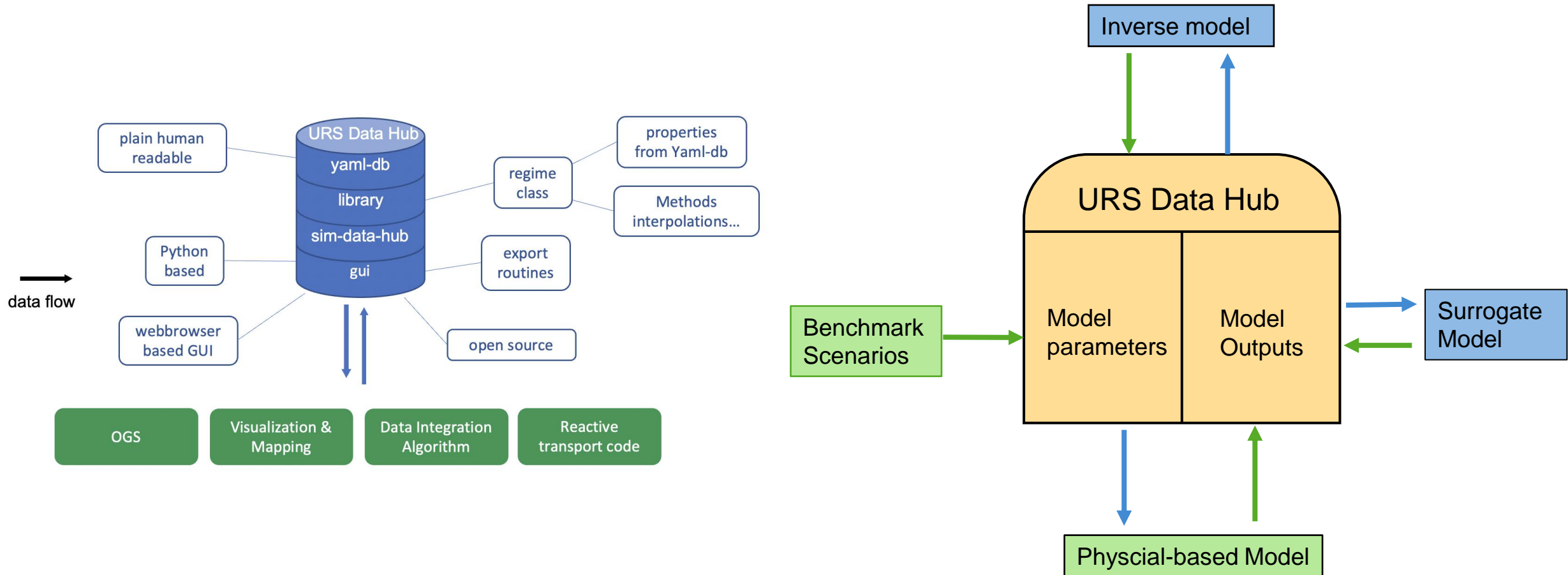
Chen Qian

Methods for Model-based Development in Computational Engineering, RWTH Aachen University, Aachen, Germany

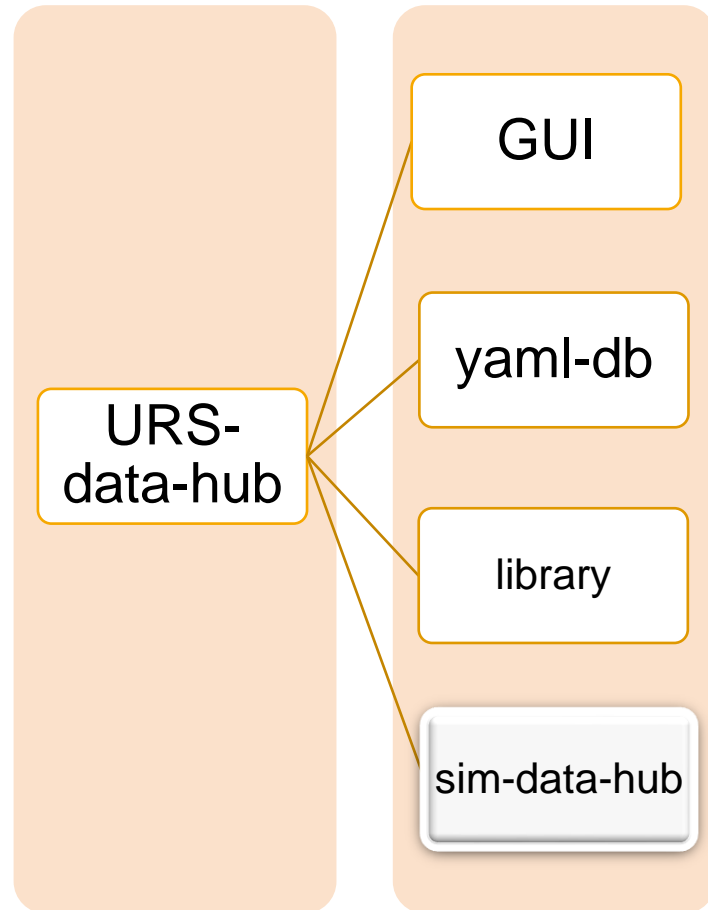
URS-data-hub: Data exchange interface



URS-data-hub: Data exchange interface



URS-data-hub



geo-fluid-dynamics / **URS-data-hub** Private Watch 1

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#) [Settings](#)

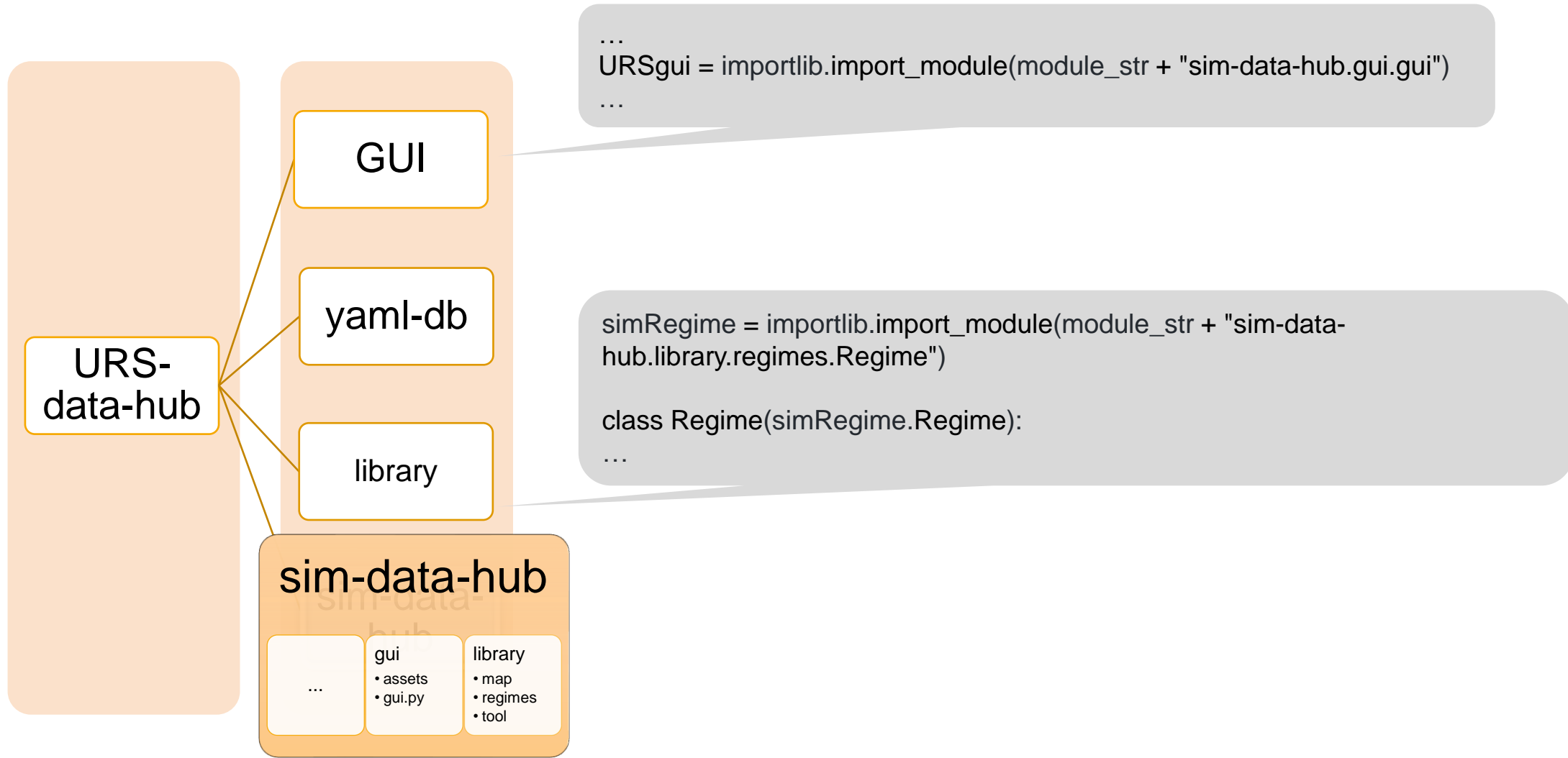
main 1 branch 0 tags Go to file Add file Code

- gui
- library
- sim-data-hub @ 6d6...**
- yaml-db

sim-data-hub

- ...
- gui
 - assets
 - gui.py
- library
 - map
 - regimes
 - tool

Sim-data-hub: submodule of URS-data-hub



Sim-data-hub: submodule of URS-data-hub and Ice-data-hub

Sim Data Hub

Chair of Methods for Model-based Development in Computational Engineering | RWTH AACHEN UNIVERSITY

Map name: Earth

Datasets: multivariables_test

Overview Map | Data | Plot

Map.

North America, Europe, Asia, Africa, South America, Australia

Drag and Drop or Select a File to Upload

Ice-data-hub

Ice Data Hub

Chair of Methods for Model-based Development in Computational Engineering | RWTH AACHEN UNIVERSITY

Planetary body: Earth

Datasets: Greenland - NorthGRIP

Overview Map | Data | Plot

Greenland - NorthGRIP

Properties: accomplishment, depth_drilled, drilling_fluid, elevation, location, project_name, thickness_ice, time_drilled

Drag and Drop or Select a File to Upload

URS-data-hub

URS

Chair of Methods for Model-based Development in Computational Engineering | RWTH AACHEN UNIVERSITY

Host rock: Claystone

Datasets: Reference Model north, claystone

Overview Map | Data | Plot

Reference Model north, claystone

Drag and Drop or Select a File to Upload

Table.

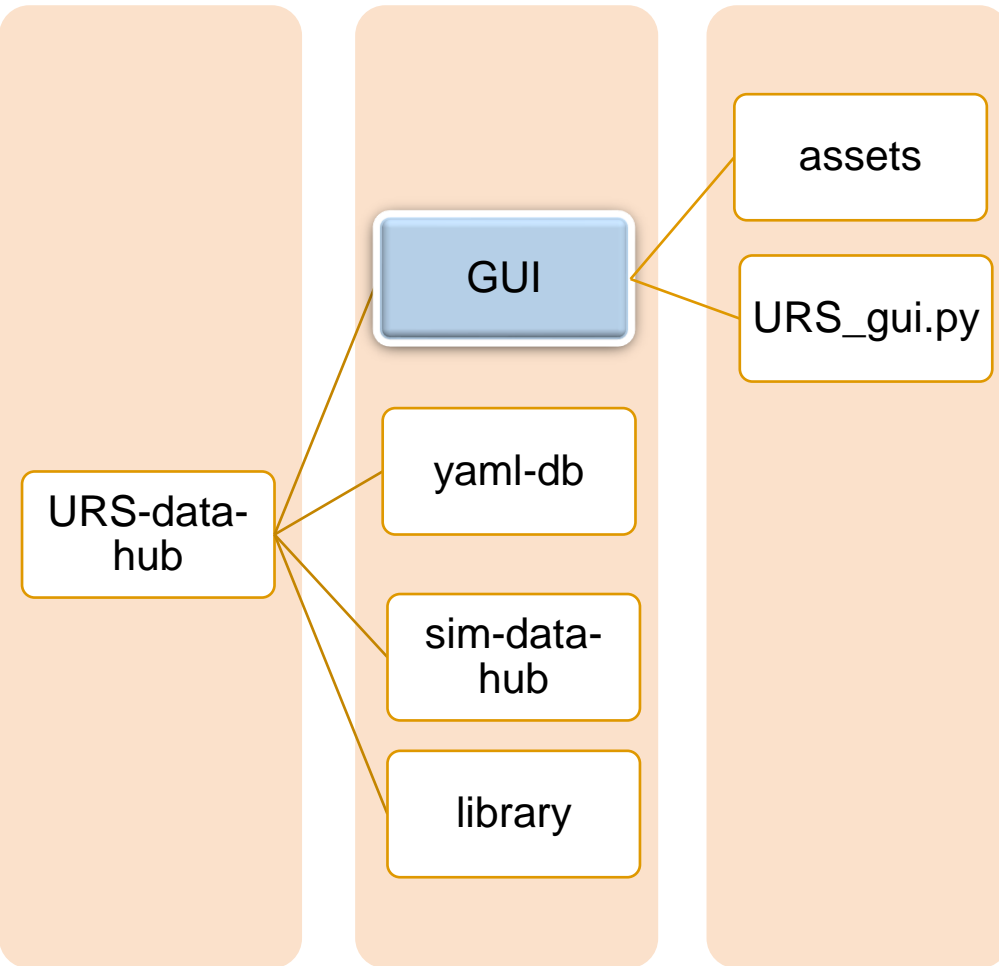
multivariables_test

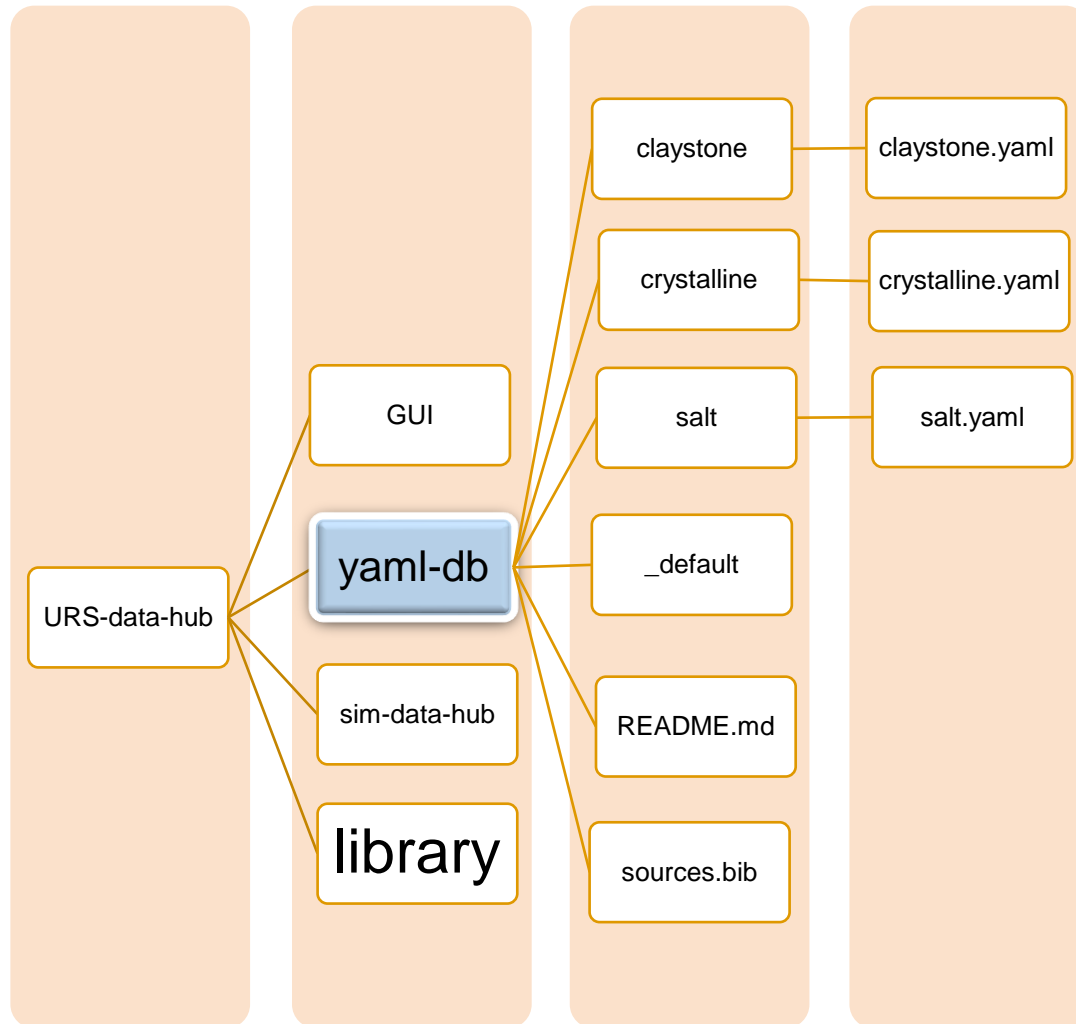
property	type	value	unit	source
test_case2	expression	9.31*x + 3.1	kg/m ³	None
test_case3	expression	9.31*x + 3.1	kg/m ³	None

Plot.

multivariables_test

URS-data-hub



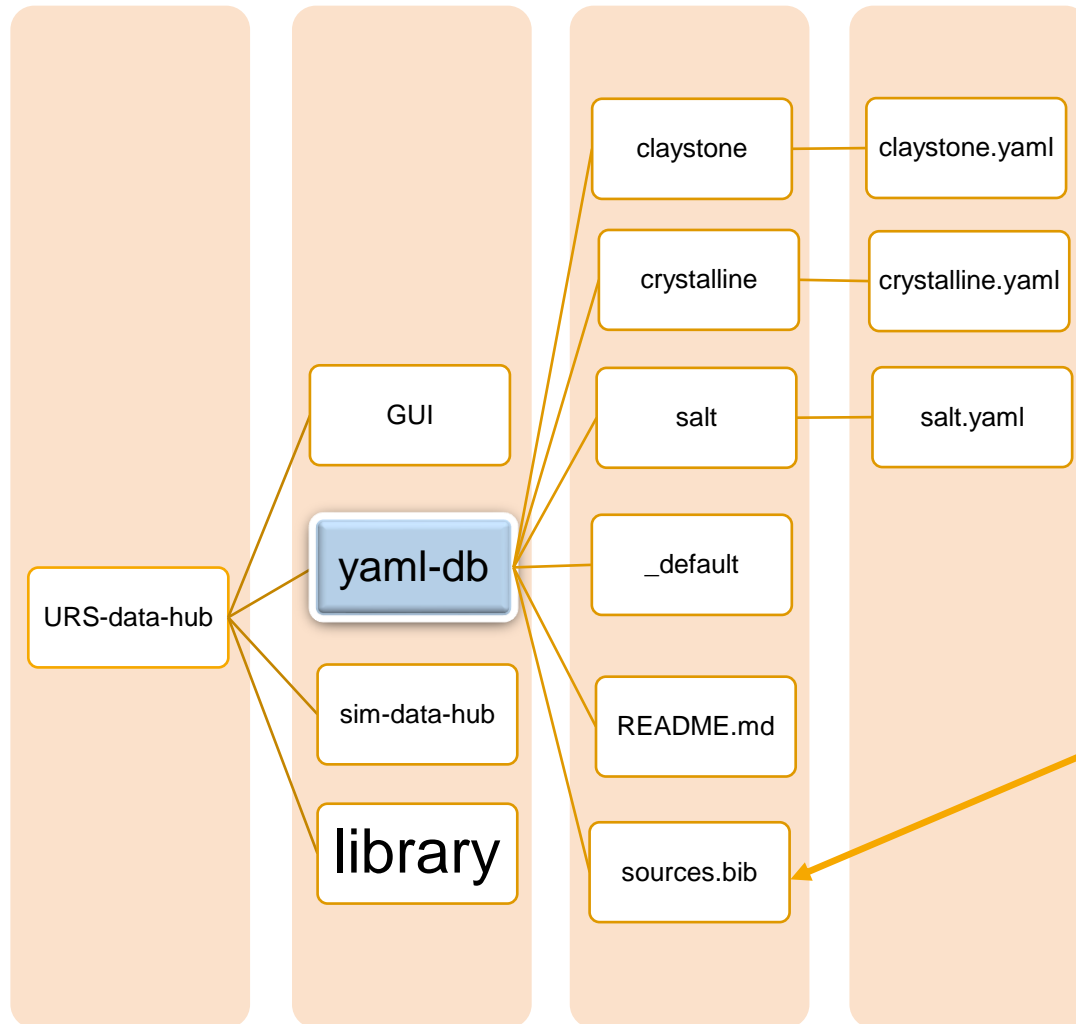


A field other than name and description has the following structure (it must contain type, value, unit and unit_str; it should contain source; other subfields can be omitted):

```

field:
  type: STR                # String out of [ scalar, array, tabulated, expression, coordinate, st
  value: VAL               # A value of type float, integer, string, array or dictionary.
  dev_pdf: STR            # Gauss or other parametrized or tabulated PDF.
  dev_value: VAL         # Hyperparameters of PDF or array with same type as value.
  unit_str: STR          # Standard string to inidate unit.
  unit: [ 0 0 0 0 0 0 0 ] # An array of the form [ kg m s K A mol cd ] that gives the unit as th
  # The SI basis units, e.g., m/s^2 is [ 0 1 -2 0 0 0 0 ].
  variable: STR          # Function argument (e.g., temperature) (must be used if type is tabul
  # expression).
  variable_unit: [ 0 0 0 0 0 0 0 ] # See above (must be used if type is tabulated or expression).
  variable_unit_str: STR # Standard string to indicate variable_unit.
  source: STR            # String with BibTeX key of data source. Free format should not be use
  meta_sys: STR         # Meta data from systematic databases, e.g. NASA database.
  meta_free: STR        # Free text meta data.
  
```

URS-data-hub



A field other than name and description has the following structure (it must contain type, value, unit and unit_str; it should contain source; other subfields can be omitted):

```
field:
  type: STR                # String out of [ scalar, array, tabulated, expression, coordinate, st
  value: VAL               # A value of type float, integer, string, array or dictionary.
  dev_pdf: STR             # Gauss or other parametrized or tabulated PDF.
  dev_value: VAL          # Hyperparameters of PDF or array with same type as value.
  unit_str: STR           # Standard string to inidate unit.
  unit: [ 0 0 0 0 0 0 0 ] # An array of the form [ kg m s K A mol cd ] that gives the unit as th
  variable: STR           # The SI basis units, e.g., m/s^2 is [ 0 1 -2 0 0 0 0 ].
  variable_unit: [ 0 0 0 0 0 0 0 ] # Function argument (e.g., temperature) (must be used if type is tabul
  variable_unit_str: STR  # expression).
  source: STR              # See above (must be used if type is tabulated or expression).
  meta_sys: STR            # Standard string to indicate variable_unit.
  meta_free: STR          # String with BibTeX key of data source. Free format should not be use
  # Meta data from systematic databases, e.g. NASA database.
  # Free text meta data.
```

2. URS-data-hub as a submodule

- Create submodule
- OGS_URS

URS-data-hub as a submodule of OGS_URS

CQVera / OGS_URS (Private)

<> Code Issues Pull requests Actions Projects

main 1 branch 0 tags

CQVera updated diffusion_sorption_decay.ipynb

URS-data-hub @ 8acce38 update from URS-data-hub

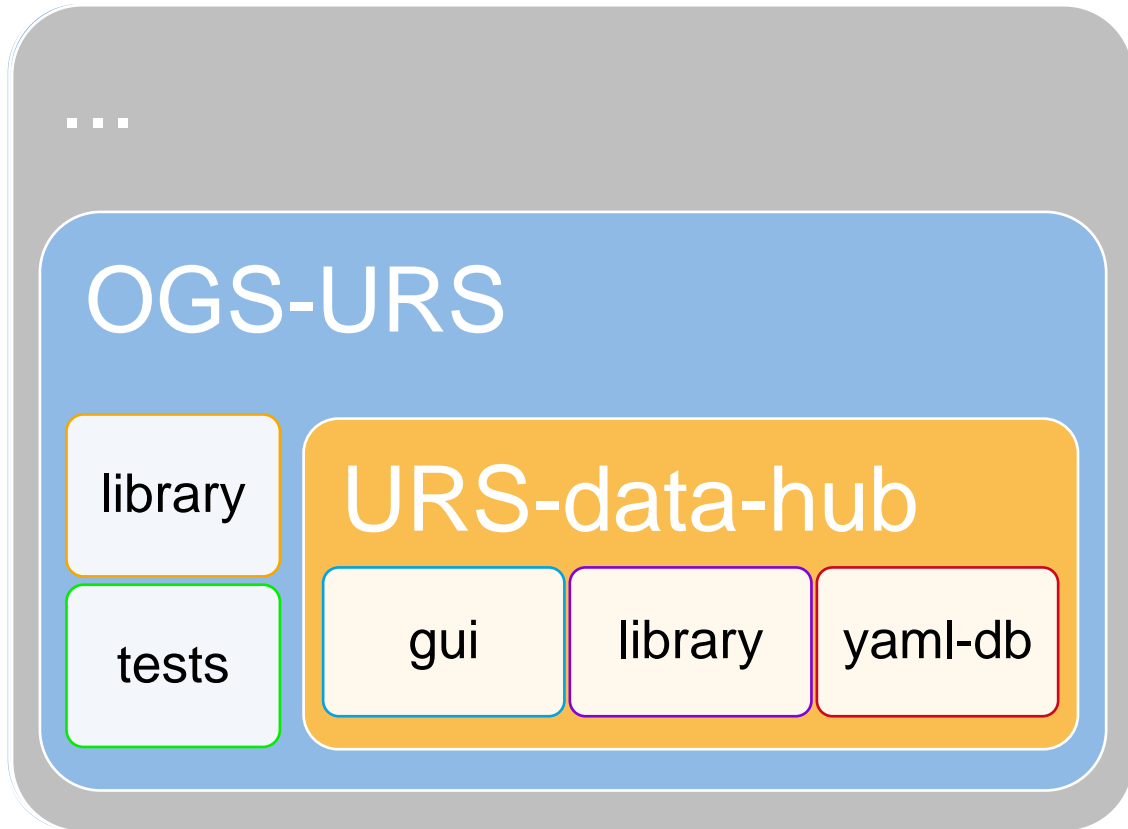
library added diffusion_sorption_decay test.

tests updated diffusion_sorption_decay.ipynb

.gitmodules added submodule URS-data-hub

README.md Initial commit

URS-data-hub as a submodule



```
1. cd OGS-URS
2. git submodule add [SSH:URS-data-hub]

# empty URS-data-hub folder
3. git submodule init
   git submodule update } or git clone --recurse-submodules [URL:URS-data-hub]

4. cd URS-data-hub

# empty sim-data-hub folder
5. git submodule init
   git submodule update } or git clone --recurse-submodules [URL:sim-data-hub]
```

Outlook

1. Create Gempy model using the URS-Data-hub

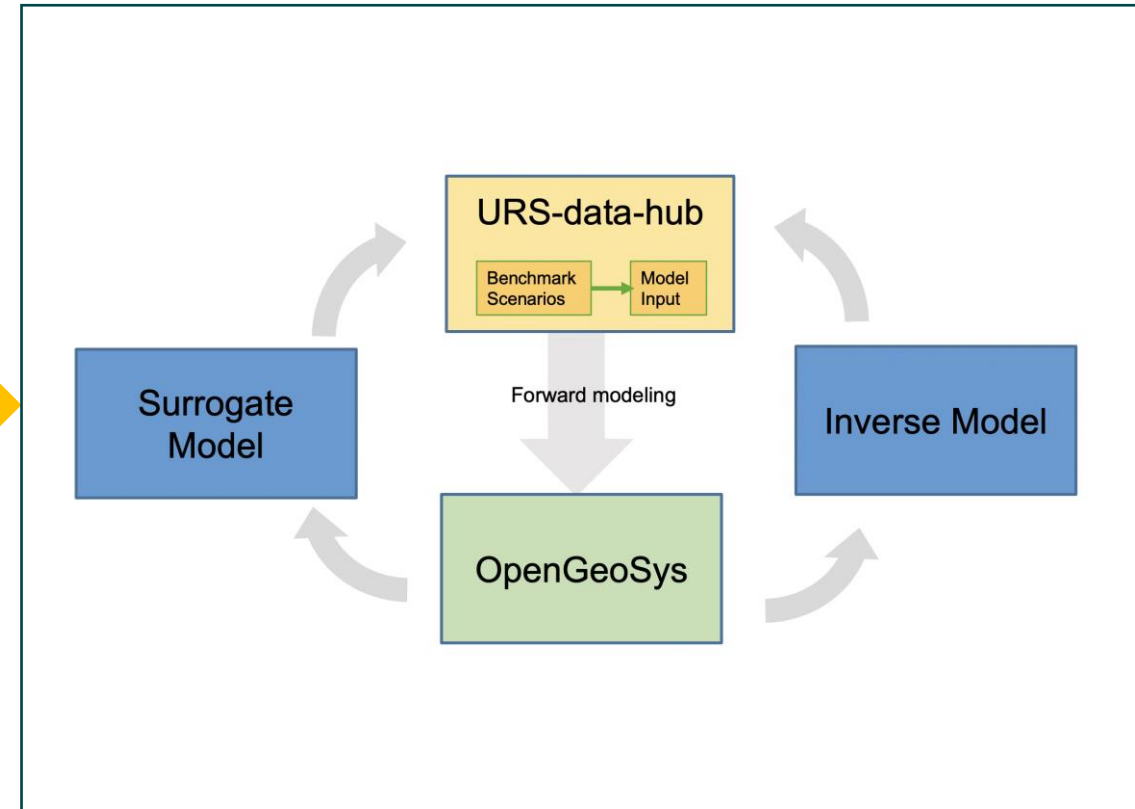
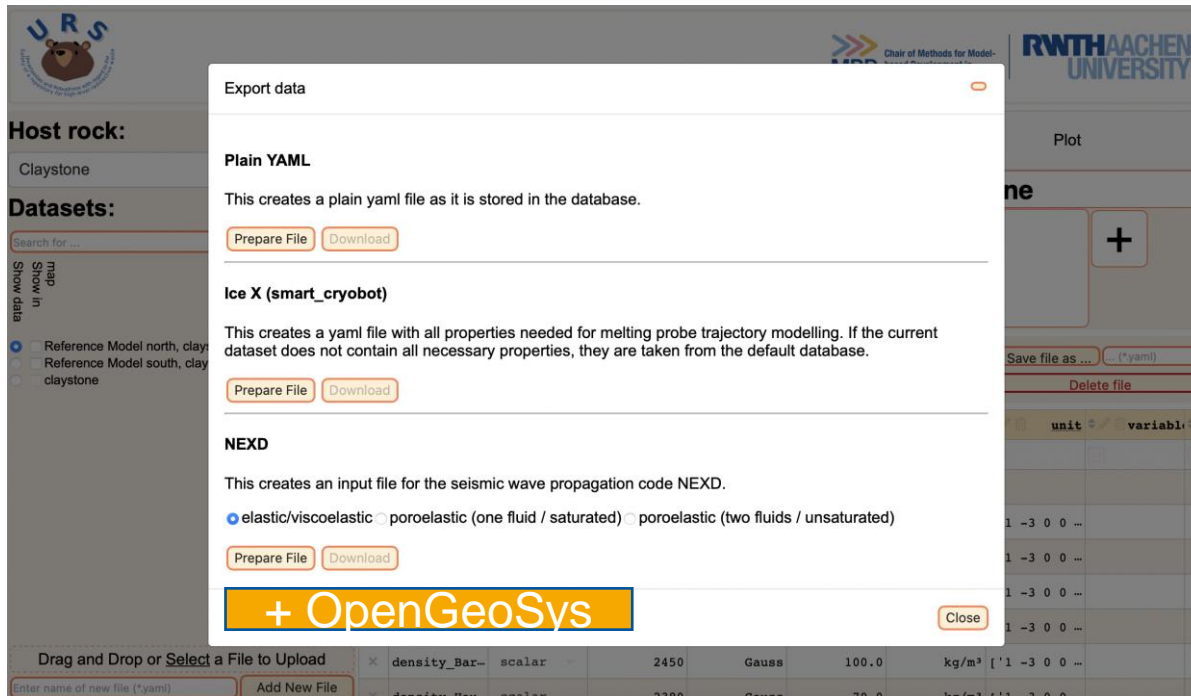


2. Use URS-data-hub as an interface between the input scenarios and OpenGeoSys.

The screenshot shows the URS-Data-hub interface with the "Export data" dialog box open. The dialog box has three sections: "Plain YAML", "Ice X (smart_cryobot)", and "NEXD". Each section has a "Prepare File" and "Download" button. The "NEXD" section has radio buttons for "elastic/viscoelastic" (selected), "poroelastic (one fluid / saturated)", and "poroelastic (two fluids / unsaturated)". A large orange button labeled "+ OpenGeoSys" is at the bottom of the dialog box. The background shows the "Host rock" section set to "Claystone" and the "Datasets" section with a search bar and a list of datasets. A table at the bottom of the background shows parameters like "density_Bar" and "density_Hau" with their respective units and values.

Outlook

2. Use URS-data-hub as an interface between the input scenarios and OpenGeoSys.



URS-data-hub

