RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

AN OPEN-SOURCE TOOLBOX FOR 3D GEOLOGICAL MODELLING IN QGIS

Nicolas Clausolles, Laure Capar, Thomas Janvier, Simon Lopez, Léana Quimerc’h

3D Geological Modelling Conference
Fremantle, 08/04/2025

Géosciences pour une Terre durable
brgm

Context

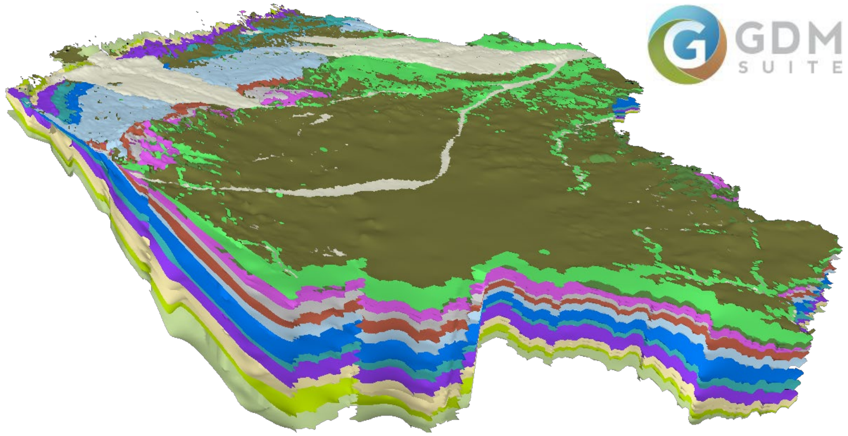
Recap from Laure's presentation

50 years of 3D geological modelling at BRGM

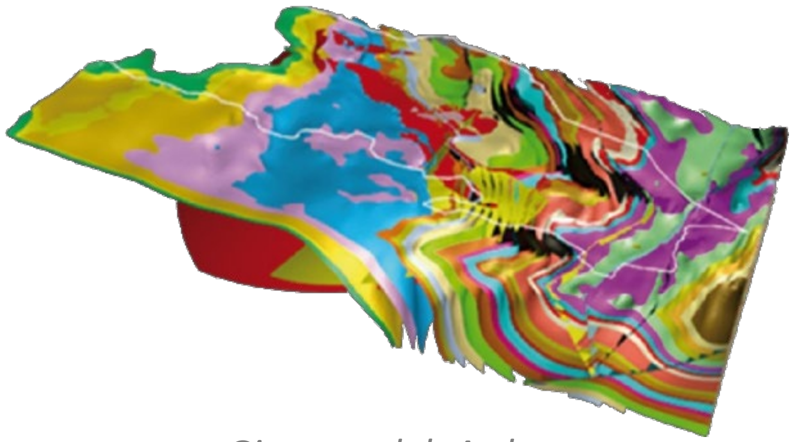
- Continuous production of multi-purpose models
  - Geological knowledge
  - Applications toward predictive geosciences

Using a variety of software

- Commercial: SKUA-GOCAD, Petrel, Isatis, Surpac, …
- Home-made:
  - GDM Suite: 2.5D modelling using elevation map kriging
  - GeoModeller: 3D implicit modelling based on potential field method
- Many "one-shot" stand-alone developments



SIGES Model – CVL province
[Tourlière, 2018 ; Badinier, 2023]



Givet model, Ardennes
[Lacquement et al., 2011]

Context & Motivations

Recapture from Laure's presentation

50 years of 3D geological modelling at BRGM

- Continuous production of multi-purpose models
  - Geological knowledge
  - Applications toward predictive geosciences

Using a variety of software

- Commercial: SKUA-GOCAD, Petrel, Isatis, Surpac, …
- Home-made:
  - GDM Suite: 2D geostatistics, 2.5D modelling using elevation maps
  - GeoModeller: 3D implicit modelling based on potential field method
- Many "one-shot" stand-alone developments

Why redeveloping from a clean state?

- Our tools implement our know-how
- Need for a "complete control" over our tools
  - Repository of our R&D
  - Basis for custom, project-oriented developments
- Home-made "innovative but legacy" solutions
  - Technical debt accumulated over decades
- New model requirements and challenges
  - Scales
  - Amount of data
  - …

Géosciences pour une Terre durable

brgm

Objectives

Specifications

Production ready

- Robust
- Fast
- Geologist-friendly

Versatile

Diversity of contexts

- Geology:
  - Scales, objects, data, …
- Projects:
  - Timed public policies projects
  - Long term research projects

Evolutive

- Repository of our R&D
- Fitted to new (unanticipated) needs / challenges

Cross-platform / Reusability

- Integration in QGIS with Windows end-users
- Compatibility with other environments
  - Web, platform, HPC, …
  - Share and exploit 3D models outside QGIS
  - Automatic building / updating frameworks

Be pragmatic: isofunctionality first!

About the overall architecture

Some design choices

2 different code layers

- Low-level programmatic layer
  - Python (by default)
  - C++ for performance bottlenecks only

+ Reusability (access computational capabilities outside QGIS)
  + Cross-platform + Evolutive (quick dev, prototyping, …)
  + Production-ready - Cross-platform - Evolutive

- High-level integration layer: QGIS
  - Exclusively about UI: just a “no-code” way to interact with the low-level layer
  - And GIS-related stuffs

+ Production-ready (integrated environment)
  + Production-ready (“geologist-friendly”)
  + Versatile (keep low-level layer about “3D”, not CRS)

Modularity

- Alternate / replace components
- Reuse components in other applications

+ Evolutive + Maintenance
+ Reusability

Géosciences pour une Terre durable
brgm

About the overall architecture (a side-step)

Reminder : 3D “structural” modelling is all about computer graphics!

Objective

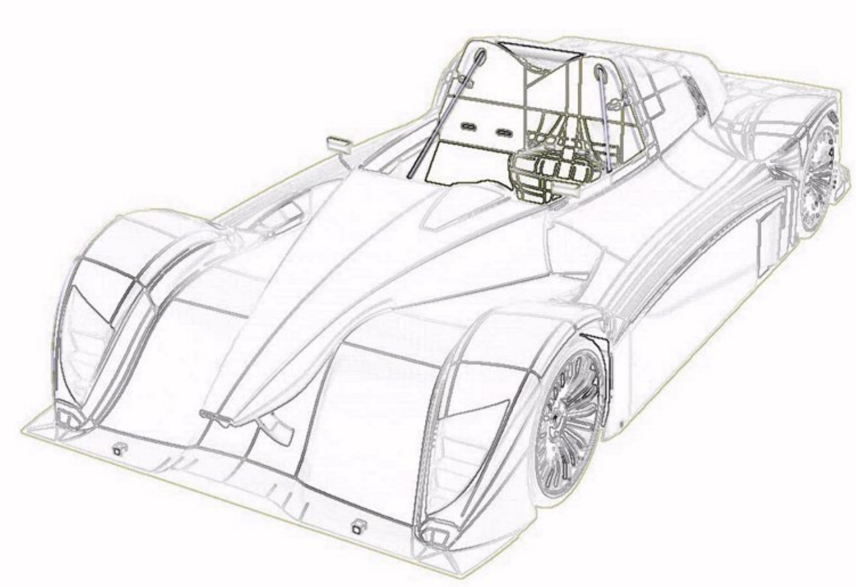
- Numerically represent the geometry of 3D (geological) volumes

B-Rep (Boundary Representation)

- Geological objects are too complex too model directly…
  - We model their boundaries (potentially many surface patches)
  - We define the objects relatively too their boundaries

Key aspect: separation of concerns

- Combinatorial: relations between surfaces, assembly
  - Model architecture
- Geometry: position of the surfaces
  - Interpolation



[Pso, via Wikimedia Commons]

About the overall architecture

Applicative architecture

QGIS layer

- 1 plugin: Forgeo

Python layer

- Description of the model
  - `forgeo`: Data structures for describing a geological model
- Interpolation
  - `gmlib`: Potential field method
  - `gdmlib`: Stacking of elevation maps
- Discretization
  - `rigs`: Data structures & algorithms for intersecting (geological) surfaces
- Integration
  - `dings`: from `forgeo` descriptions, setup `gmlib`/`gdmlib` interpolators and setup `rigs` internal data structures

Géosciences pour une Terre durable

Forgeo: the model description

Main elements of a geological model

Stratigraphic column

- Elements to model (units & stratigraphic interfaces)
- Depositional logic (architecture of the stratigraphic deposits)

Stratigraphic/Layered model

- Data assignation to column elements
- Parametrization of the interpolations
- Selection of the discontinuities affecting the elements

Fault network

- Faults that can be added to the model
- Relations between faults (“stops on”)
- Parameterization of the interpolations

Forgeo: the model description

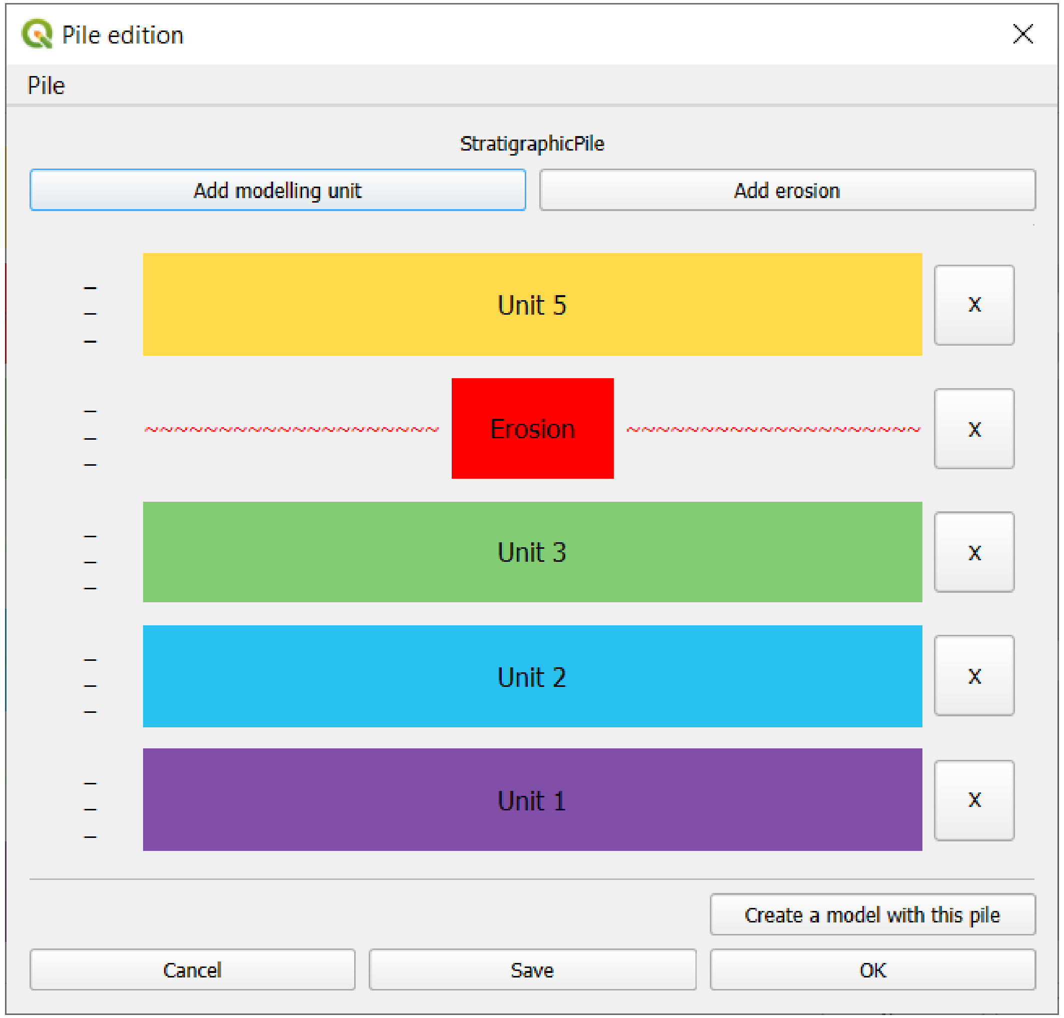
Stratigraphic pile: the architecture

What are the elements to model?

- Modelling units (any *-stratigraphic unit)
- (Stratigraphic) interfaces

How are they organized?

- Depositional logic:
  - Some matter was removed → Erosion
  - Simple “piling” of the units → Onlap



Forgeo: the model description

Stratigraphic pile: the architecture

What are the elements to model?

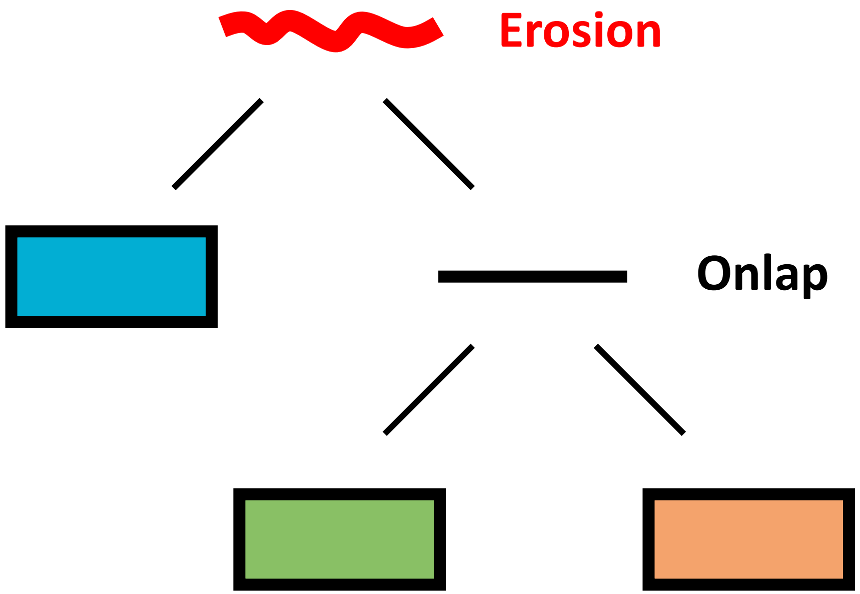
- Modelling units (any *-stratigraphic unit)
- (Stratigraphic) interfaces

How are they organized?

- Depositional logic:
  - Some matter was removed → Erosion
  - Simple "piling" of the units → Onlap

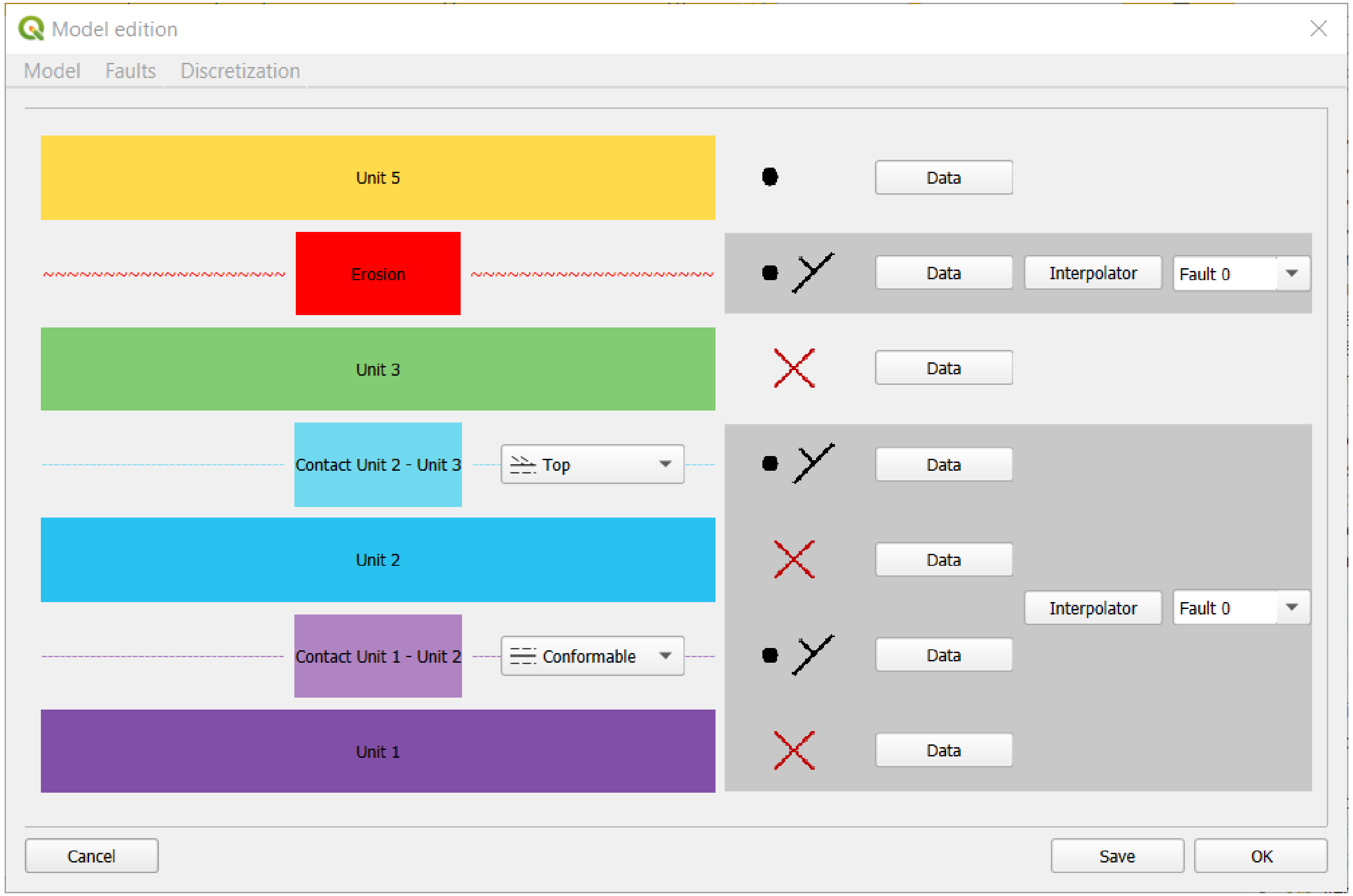
Model architecture

- Intersection rules between model surfaces
- (Some of the) Combinatorial part of the B-Rep
- Specificity: we manipulate semi-infinite surfaces



Forgeo: the model description

Stratigraphic model: the geometry



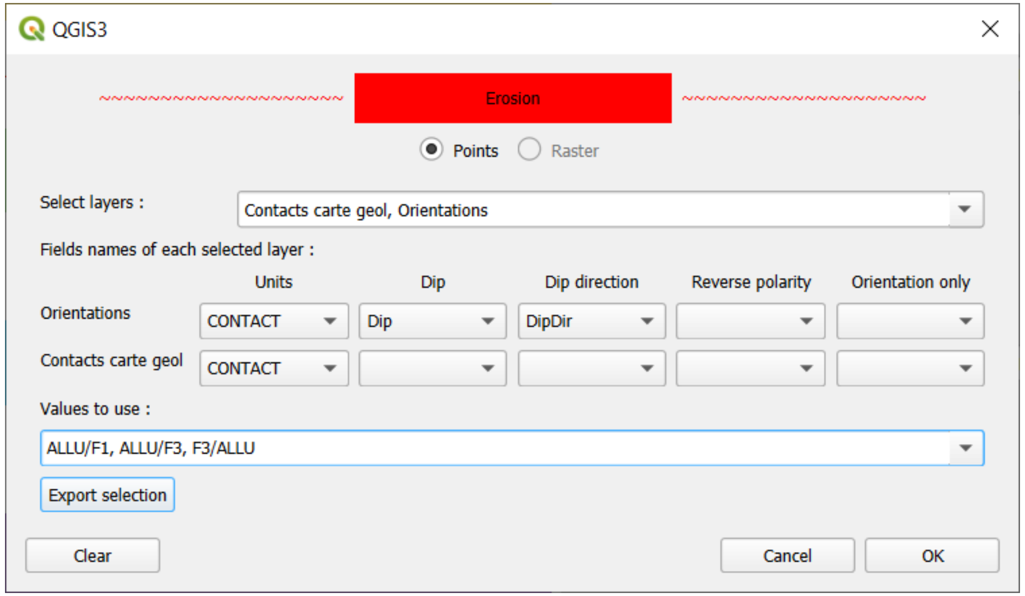
Forgeo: the model description

Stratigraphic model: the geometry

Data assignation

- Assign per model element
  - Modelling unit
  - Stratigraphic interface

- 2 types of data
  - Observations: the element exists at (x, y, z)
    - Units: inequality constraints, …
    - Interfaces: contact point
  - Orientations: at (x, y, z) the “principal” orientation of the element is $\vec{v}$
    - S0 of a unit
    - Dip of an interface
    - Can be used or not as observations data too



Géosciences pour une Terre durable
brgm

Forgeo: the model description

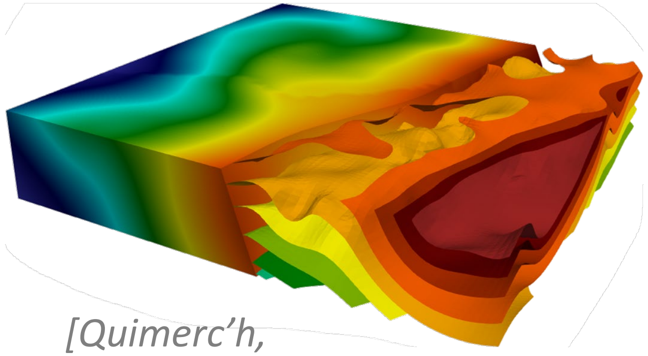
Stratigraphic model : the geometry

Interpolation parametrization

Basis functions

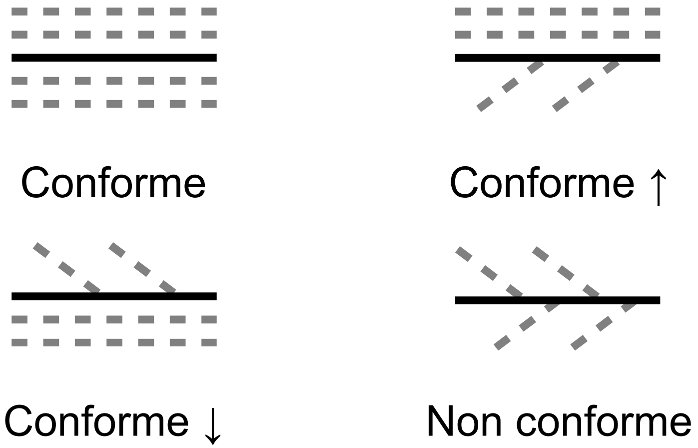
- Currently: geostatistical methods
  - Elevation maps (2D universal kriging): $z(\boldsymbol{x}) = \boxed{\sum_{i=1}^{I} \lambda_i \chi_i(\boldsymbol{x})} + \boxed{\sum_{l=1}^{L} d_l p_l(\boldsymbol{x})}$
  - Potential field method (3D cokriging): $f(\boldsymbol{x}) = \boxed{\sum_{i=1}^{I} c_i \chi_i(\boldsymbol{x})} + \boxed{\sum_{l=1}^{L} d_l p_l(\boldsymbol{x})}$

Trend functions



[Quimerc'h, 2023]

- Information to provide
  - Variogram model
  - Drift terms
  - (Neighborhood)
- Notion of “conformity” in implicit modelling
  - Non-intersecting surfaces can be represented by a single scalar field
  - For each surface: its relations with the above and below units
- Discontinuities
  - Affecting each interpolator

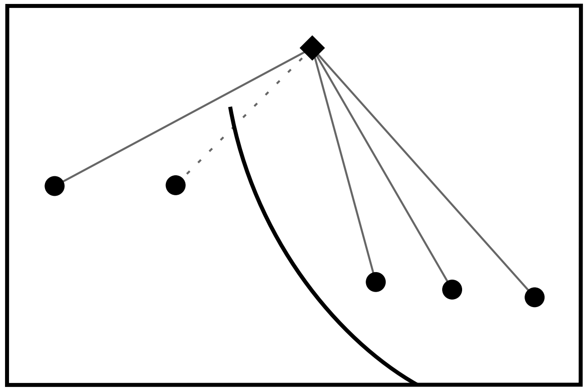


gmlib / gdmlib: interpolations

"On the fly" interpolators

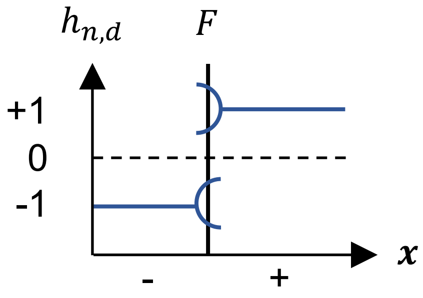
gdmlib

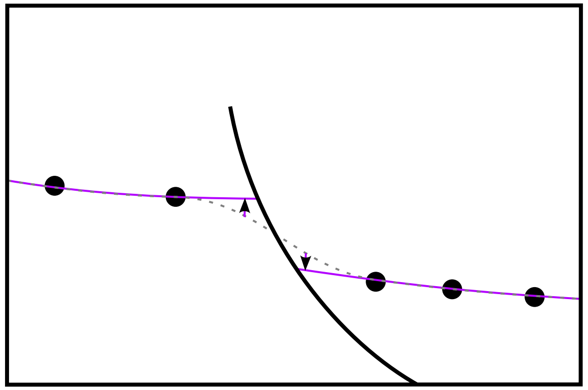
- 2D kriging
  - moving neighborhood with fault screens
- Based on gstlearn *[MINES Paris - PSL University, 2025]*
  - Core in C++



gmlib

- 3D co-kriging
  - Variable + its gradient
  - Dual formulation with drifts for faults
- C++ for performances





Géosciences pour une Terre durable
brgm

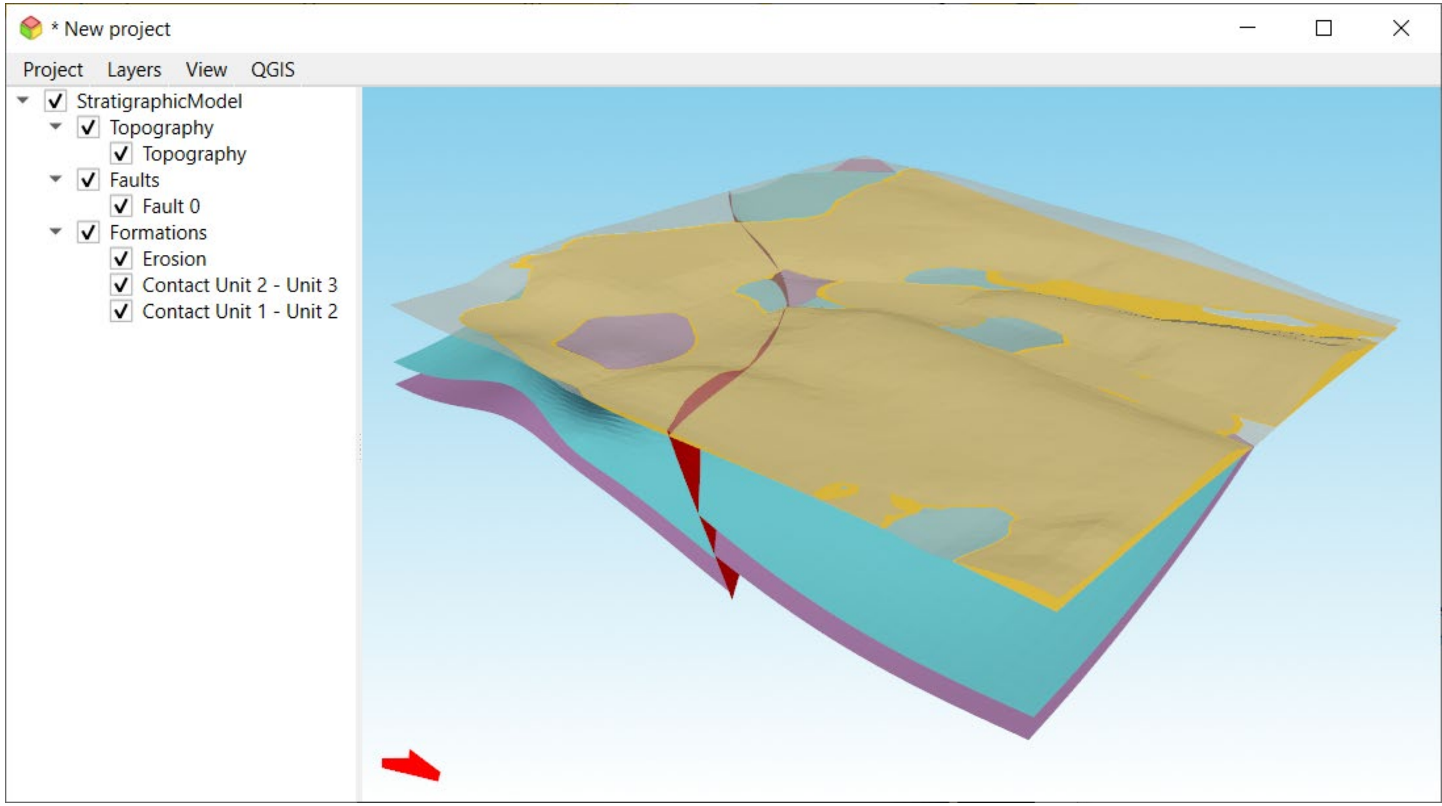
rigs: discretizing the model

Robust intersection of geological surfaces

Generic discretization for implicit B-Reps

[Du et al., 2023]

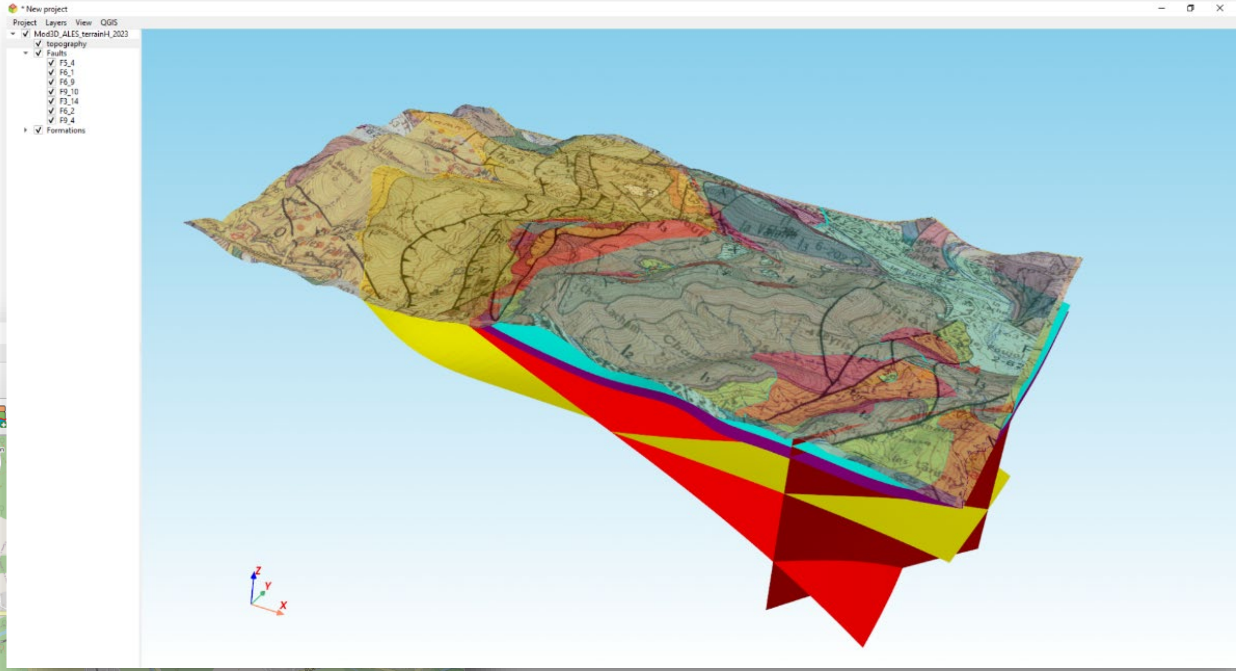
- Space partitioning trees
  - Stratigraphic interfaces
  - Discontinuities
- Exact intersections
  - Per-cell evaluation of all intersections
  - Watertight representations
  - Exact-evaluations on discontinuities
- Outputs:
  - Surfaces & volumes
  - Intersection with any surface, line, …
- C++ for performances

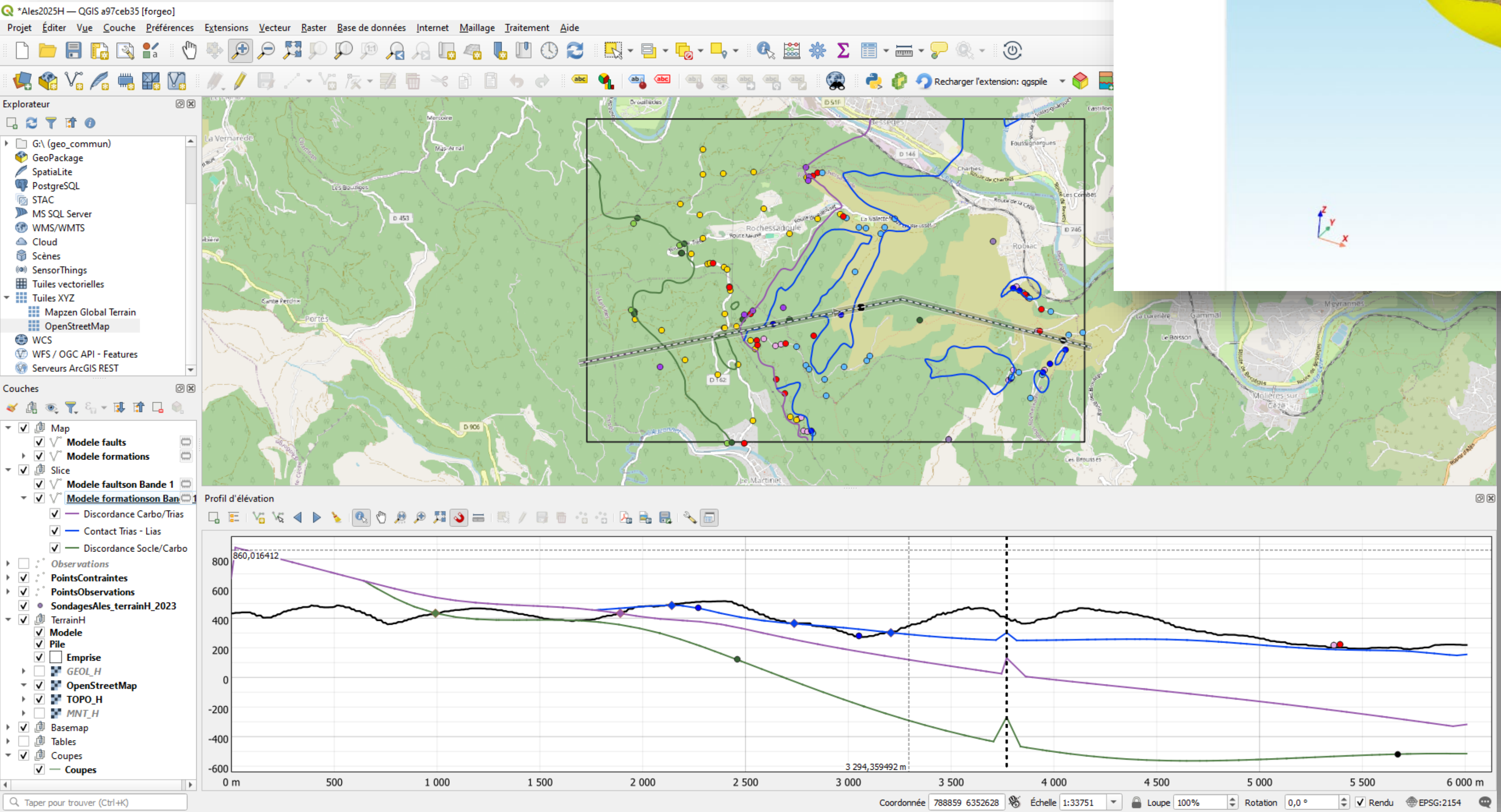


What a wonderful software!

We often share what we are proud of…

Feedbacks about difficulties is also enriching





Géosciences pour une Terre durable

brgm

The devil lies in the details…

Modularity & micro-services

For 2 years: 2 packages

- Generic serialization package for “common” data structures
- Geomodelling-related data structures

Since last week

- One single package

Why?

- KISS : Genericity is great, but no need to factorize while we use it at a single place
- Requires more care:
  - Keep dependencies up-to-date
  - More stuffs to install
- More complex to dive into for external developers

The devil lies in the details…

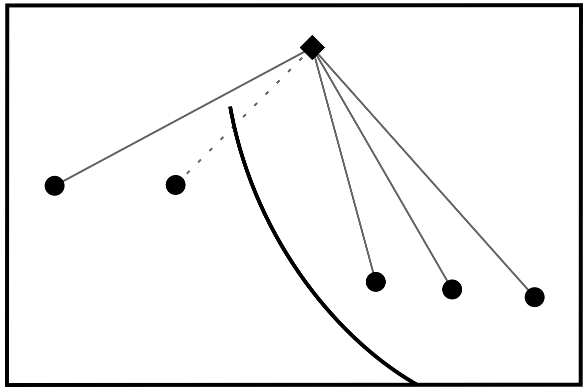
Combining multiple modelling methods in a single model

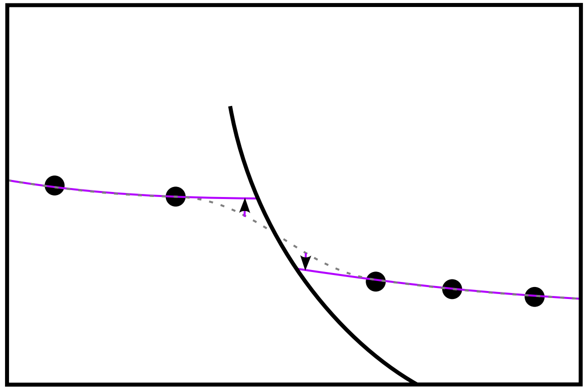
Starting point

- Different structures may have very different data
- A single method may not be suited to all model elements

Requirements for a proper model discretization

- Common representation
  - Implicit modelling > elevation maps are explicit modelling…
- Fully 3D approach
  - Times explode for “2D interpolation”
- Handling faults
  - Commonly, they affect several model elements
  - Fault interpolation shall fit both method requirements
    - 2D polyline defining a vertical plane
    - Two 3D functions: fault surface + throw



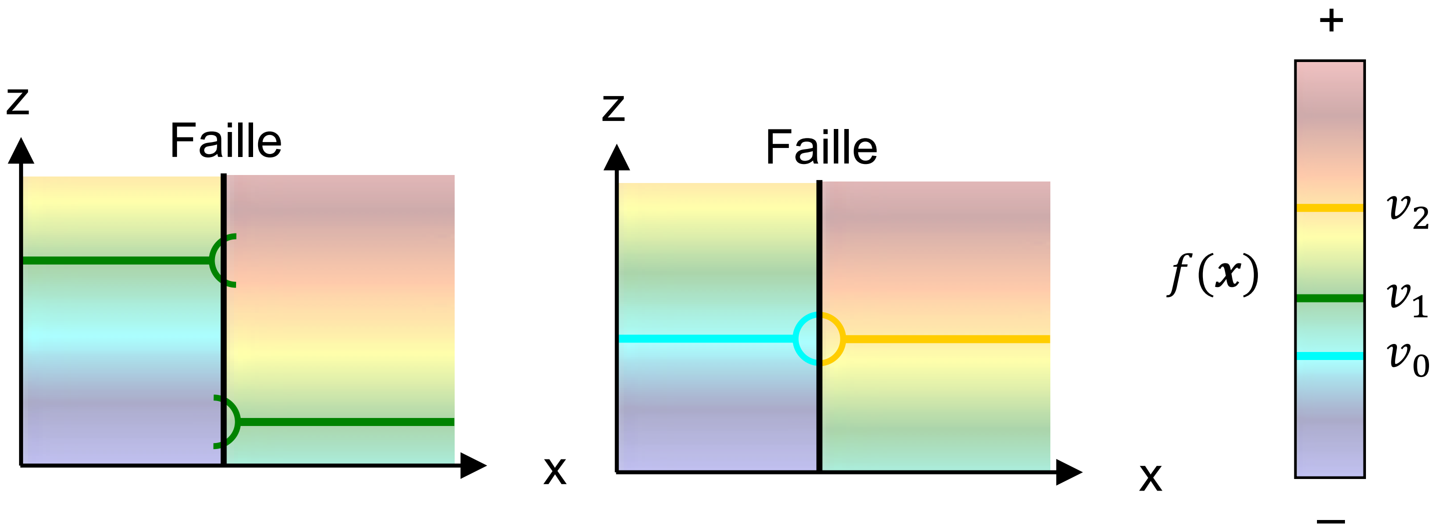


The devil lies in the details…

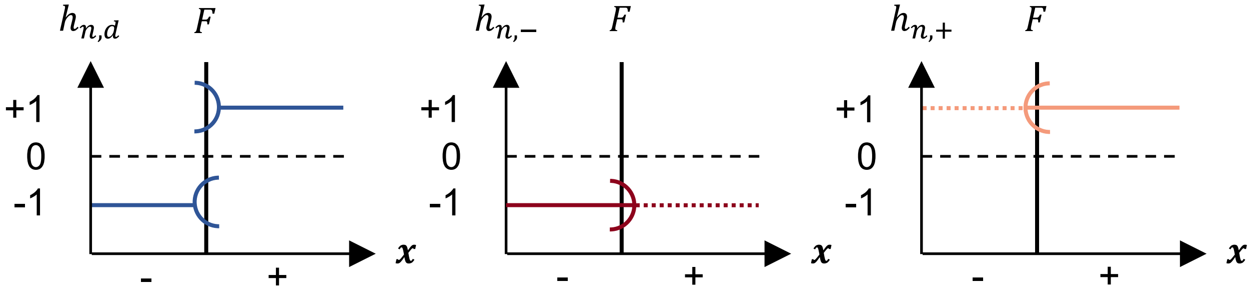
The impact of design choices

Exact surface intersections

- Require
  - Evaluate on discontinuities, twice
  - Require specific evaluation schemes…
  - … to avoid numerical issues



- Implication on interpolators
  - Forces libraries to implement “sided-evaluation”
  - gmlib: “simply” change the drift sign
  - gdmlib: rebuild an interpolator from scratch…



Géosciences pour une Terre durable
brgm

The devil lies in the details…

The price of performances

From python…

- High-level interpreted language
  - Cross-platform (interpreter)

- Performances ?
  - Natively inexistant, not meant for it
  - Tools exist:
    - numpy, scipy, ...
  - Not suited to any problem

… To C++

- Low-level compiled language
  - “Single” platform

- “Ensures” efficient codes
  - Memory usage
  - Compiler optimizations

- Drawback: code is much more complex
  - To implement
  - To maintain
  - To deploy

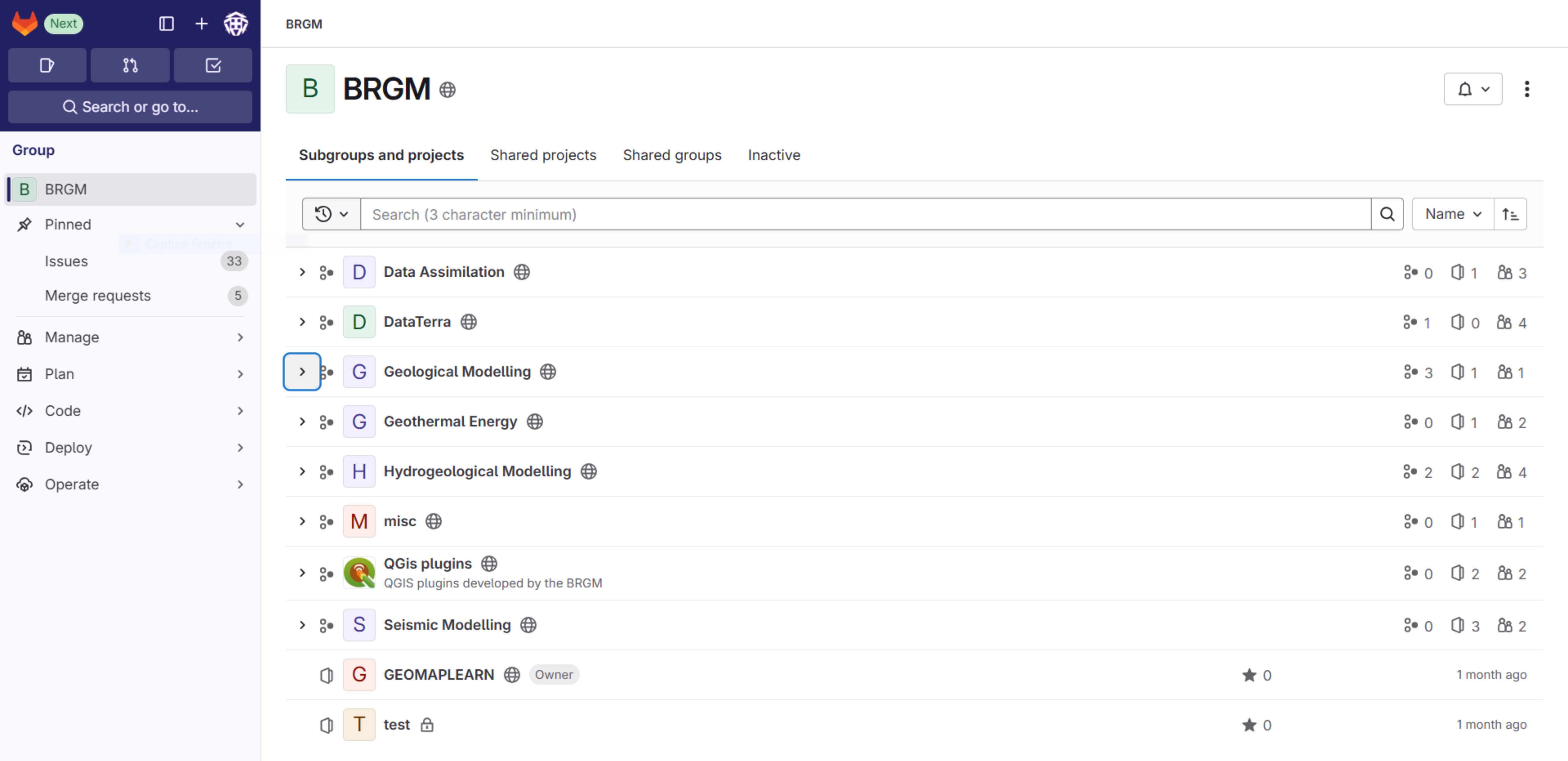
Conclusion

Multi-purpose 3D modelling

- Focus on our daily production needs
  - Multiple environments, scales and data
- Evolutive
  - Repository of our R&D
  - On-demand dev for specific projects needs
- Trade offs are mandatory
- Finally releasing!
  - First training session last week
  - Used in production this year

Developing QGIS plugins

- “Natural” target for geomodelling projects
  - We all move to it
  - GIS already integrates all (most?) geological data
- Need for building up “3D geomodelling” community!
  - Same needs and difficulties
  - QGIS is originally not suited for 3D
  - 3D people represent a “negligible” fraction of GIS users
- Open-source
  - https://gitlab.com/brgm
    - Geological Modelling
    - QGIS plugins



https://gitlab.com/brgm

Thank you for your attention!

Acknowledgements

ANR-22-EXSS-0005:
PEPR Sous-Sol PC4 Digital Earth

Géosciences pour une Terre durable
brgm