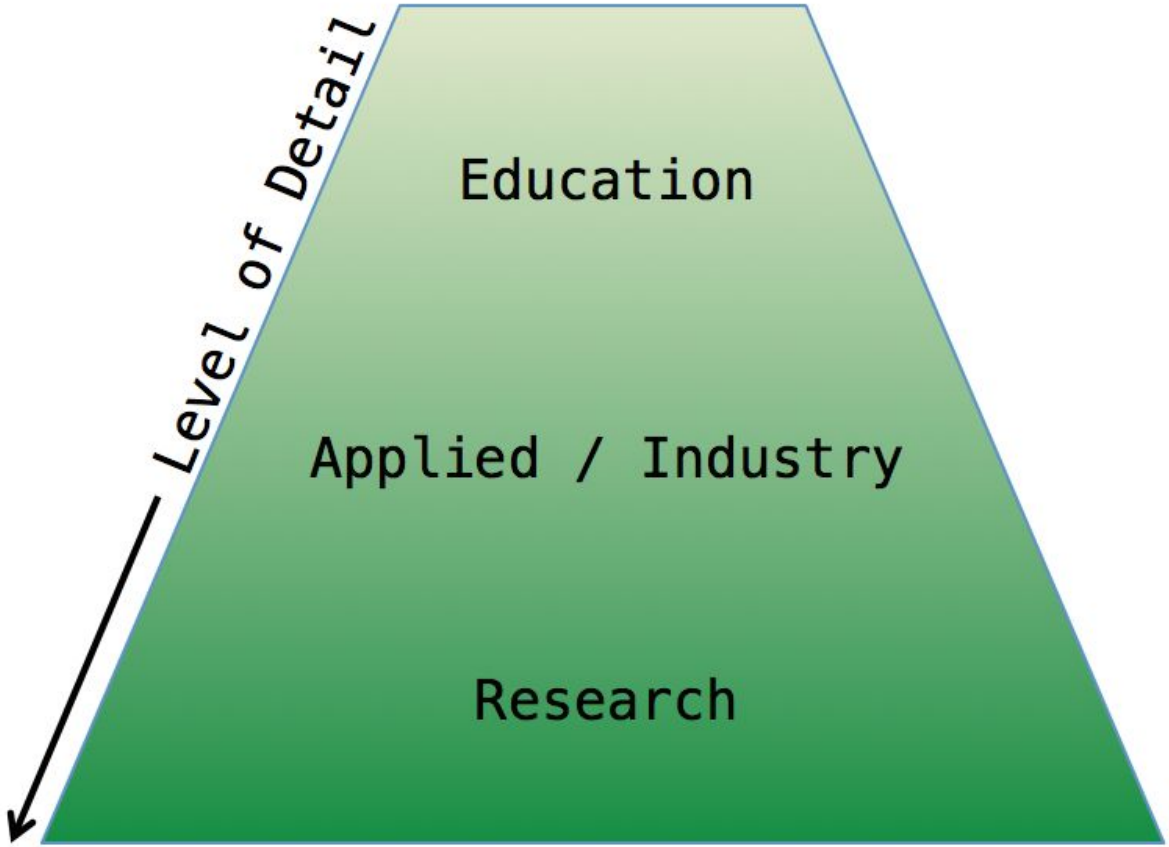
Using Python to span the gap between education, research, and industry applications in geophysics

Lindsey Heagy
& Rowan Cockett, Gudni Rosenkjaer, Seogi Kang, Doug Oldenburg, et al.

Geophysical Inversion Facility
University of British Columbia

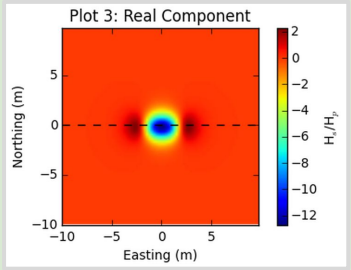

Level of Detail
Education
Applied / Industry
Research

Education

param 1

param 2

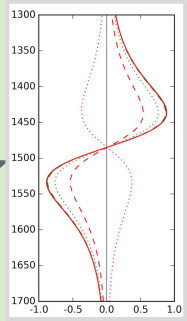
System



Industry & Applied

```
>>> import SimPEG
>>> SimPEG.solveMyProblem(someParameters)
```

Box that solvesMyProblem



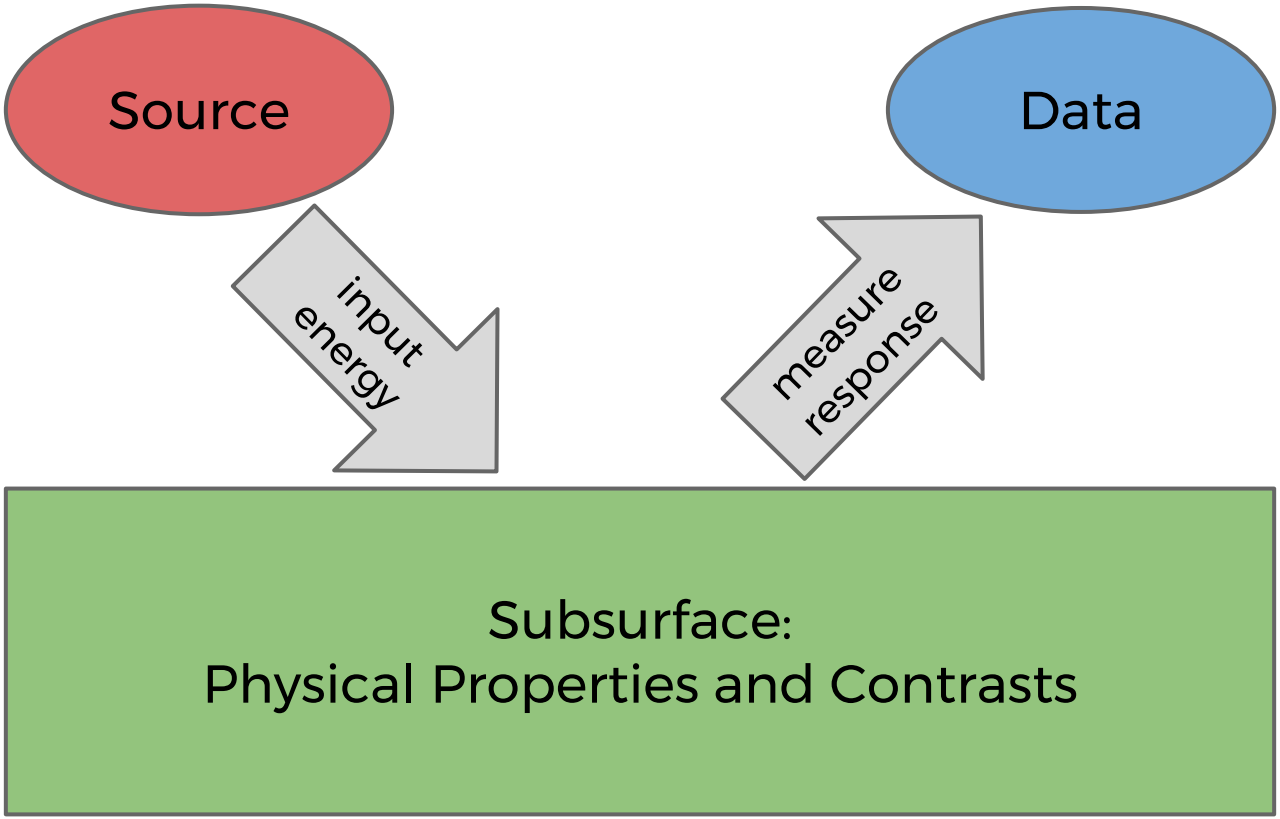
Research

Question

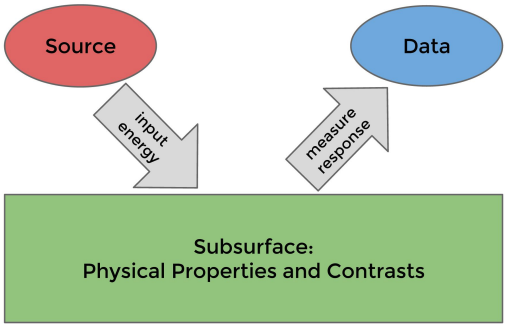
Box with details

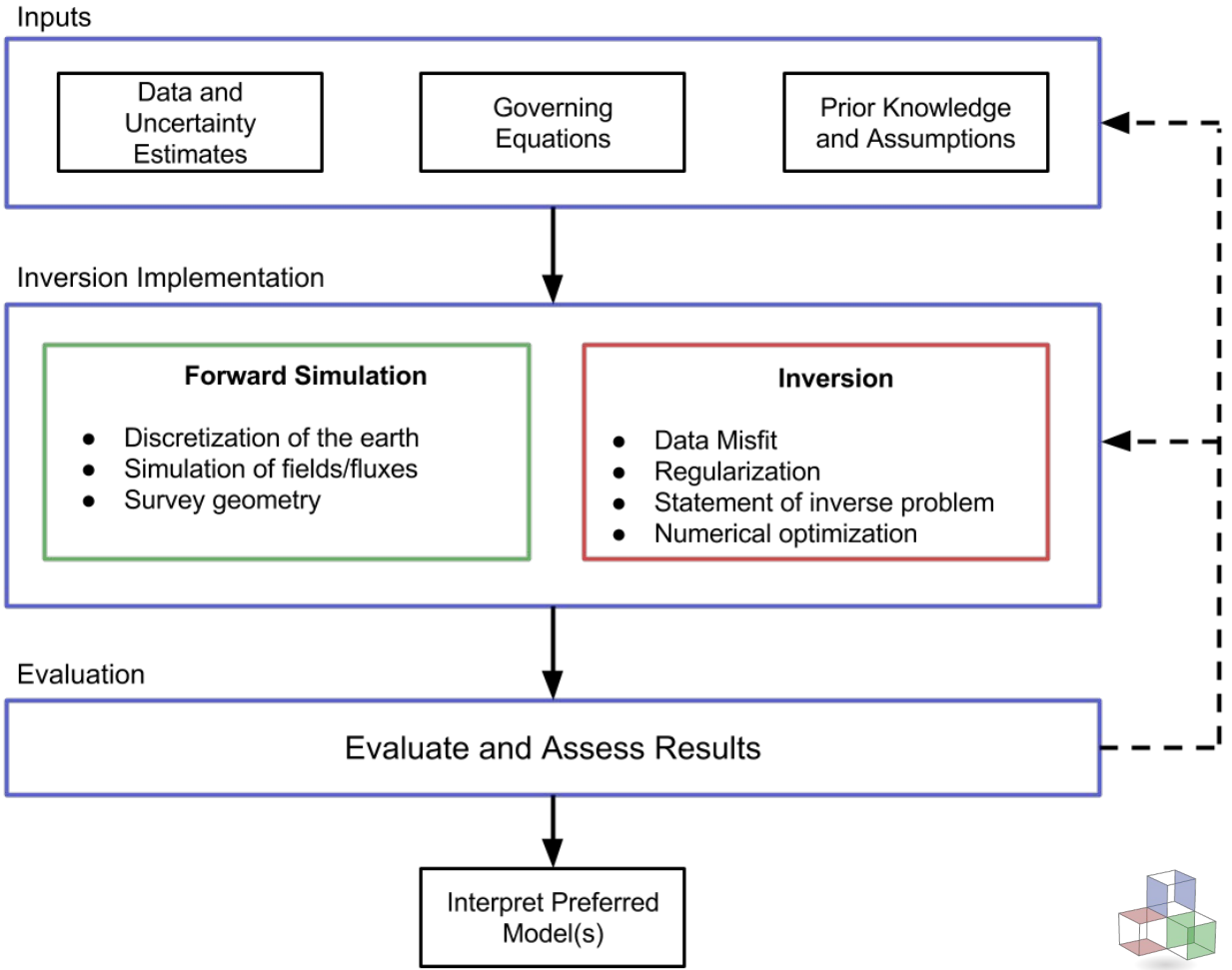
PhD?

Geophysics!


Source
Data
input energy
measure response
Subsurface:
Physical Properties and Contrasts

Geophysics!

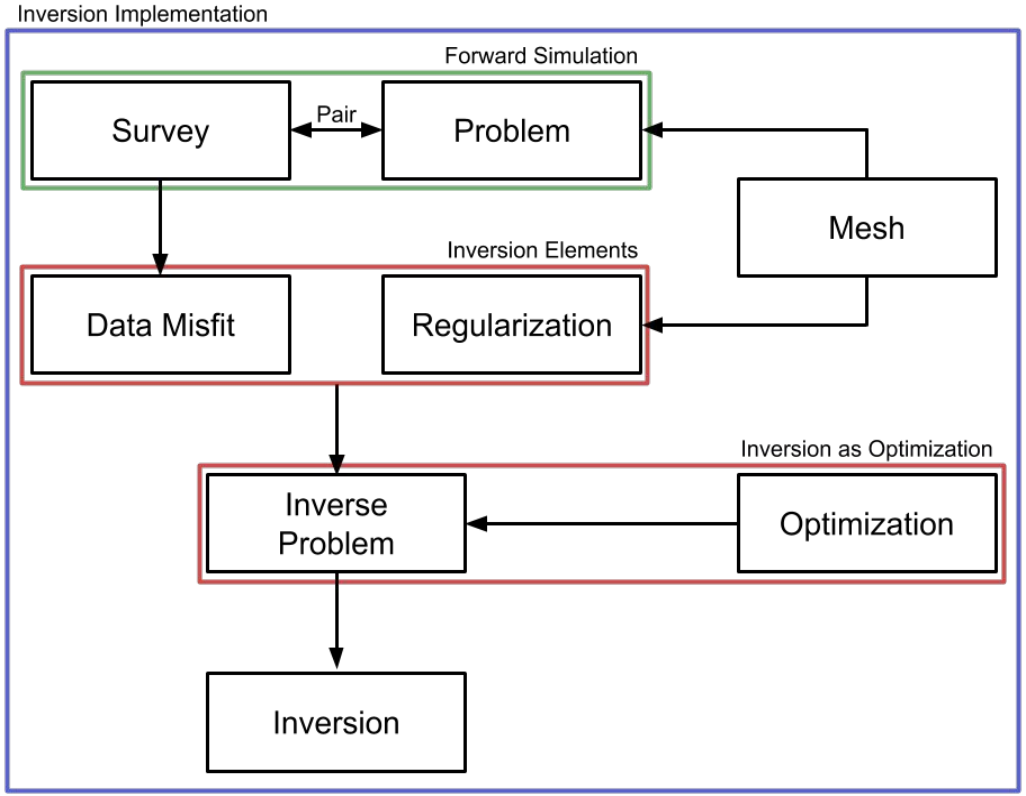

Source
Data
input energy
measure response
Subsurface:
Physical Properties and Contrasts


Inputs
Data and Uncertainty Estimates
Governing Equations
Prior Knowledge and Assumptions
Inversion Implementation
Forward Simulation
Discretization of the earth
Simulation of fields/fluxes
Survey geometry
Inversion
Data Misfit
Regularization
Statement of inverse problem
Numerical optimization
Evaluation
Evaluate and Assess Results
Interpret Preferred Model(s)

How?

- Framework
- Modularity

- Testing
- In the open.



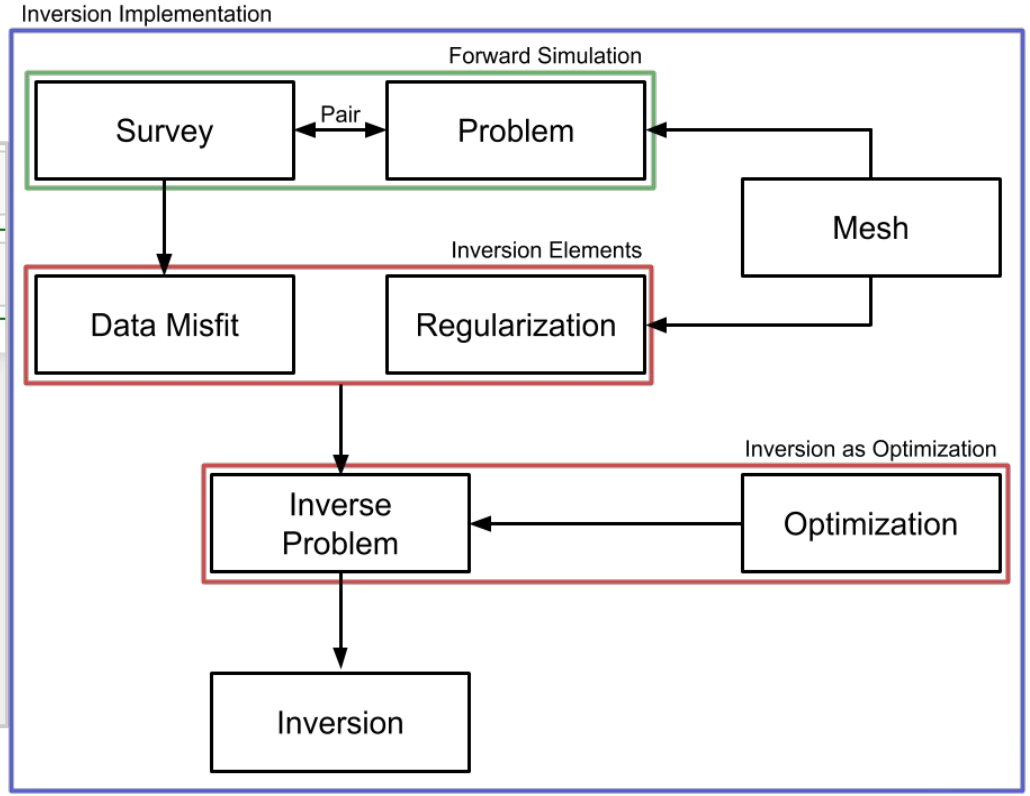
simpeg

pypi v0.1.3 | downloads 3k/month | license MIT | build passing | coverage 83%

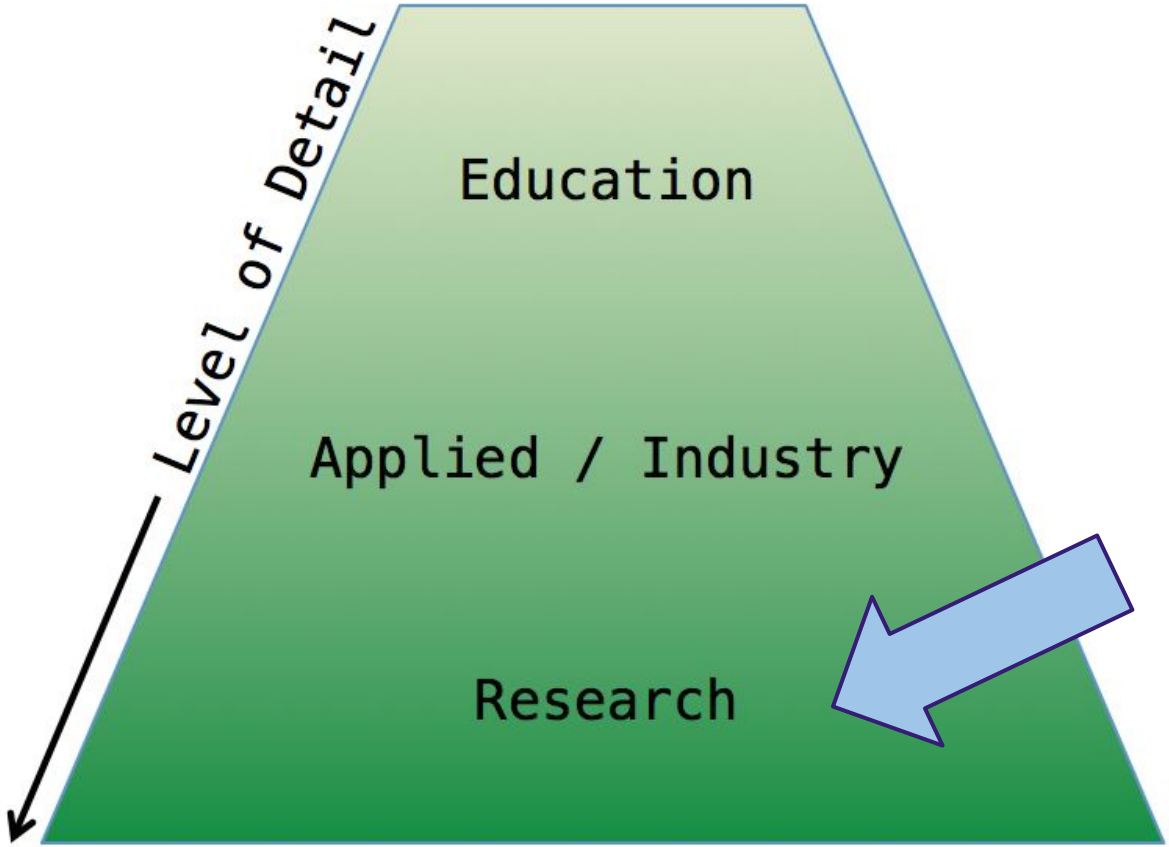
How?

```
In [1]: import SimPEG

In [ ]: SimPEG.
```

```
SimPEG.DataMisfit
SimPEG.Directives
SimPEG.Fields
SimPEG.InvProblem
SimPEG.Inversion
SimPEG.Maps
SimPEG.Mesh
SimPEG.Models
SimPEG.Optimization
SimPEG.Problem
```

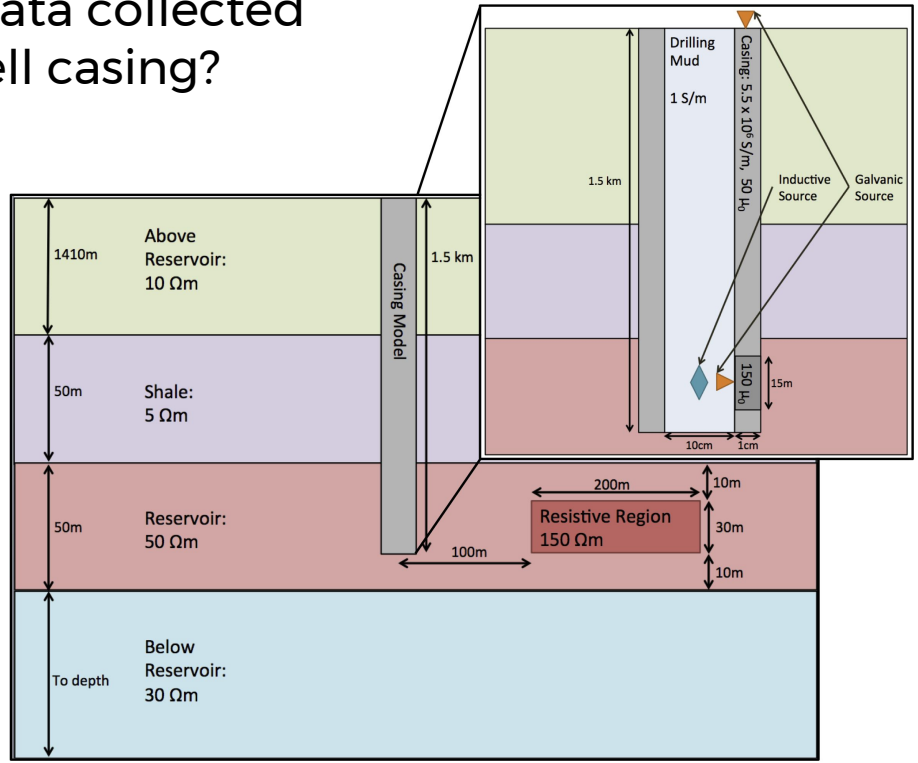


pypi v0.1.3 | downloads 3k/month | license MIT | build passing | coverage 83%


Level of Detail
Education
Applied / Industry
Research

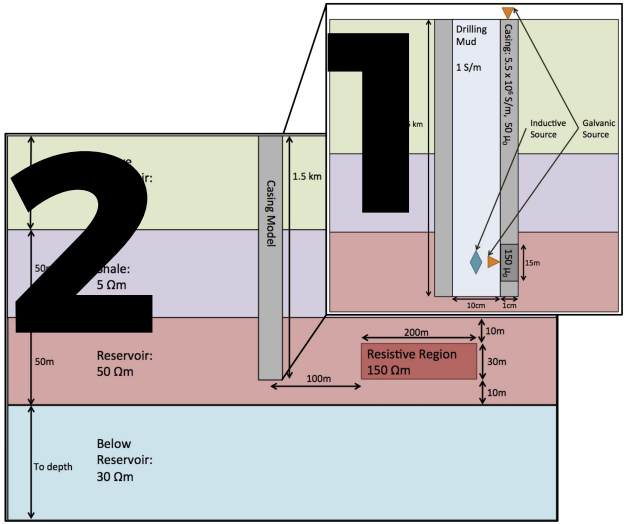
Research

Problem: How do we model and invert electromagnetic data collected in settings with well casing?



Research

Approach: Split it into two problems using primary-secondary



Details...

- Multiple variable physical properties
- Primary problem: 2D problem with 3D fields
  - problem formulation

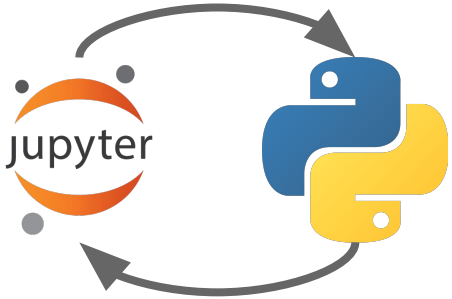
```
In [ ]: EM.FDEM.
        EM.FDEM.ProblemFDEM_b
        EM.FDEM.ProblemFDEM_e
        EM.FDEM.ProblemFDEM_h
        EM.FDEM.ProblemFDEM_j
```

- Secondary Problem: source depends on the model
  - need derivatives in inverse problem
- ...

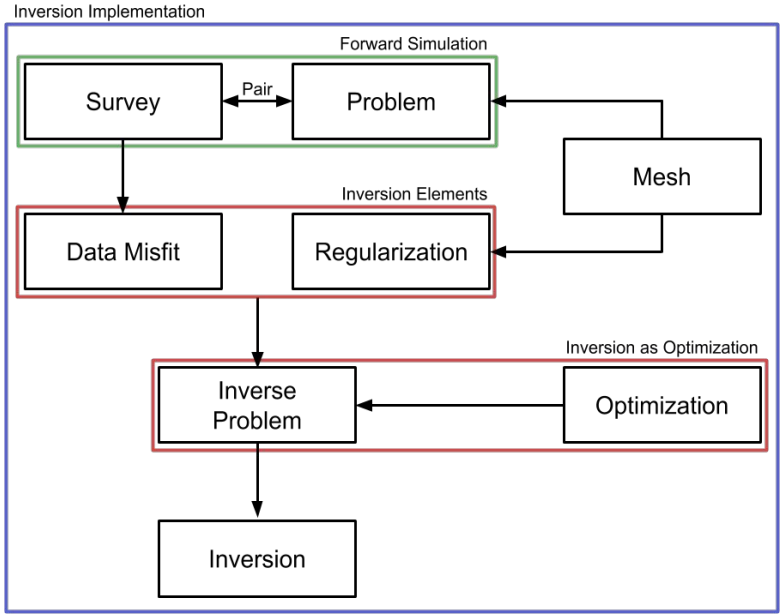
Research

Resources & Practices:

Iterative Development



Modular Framework

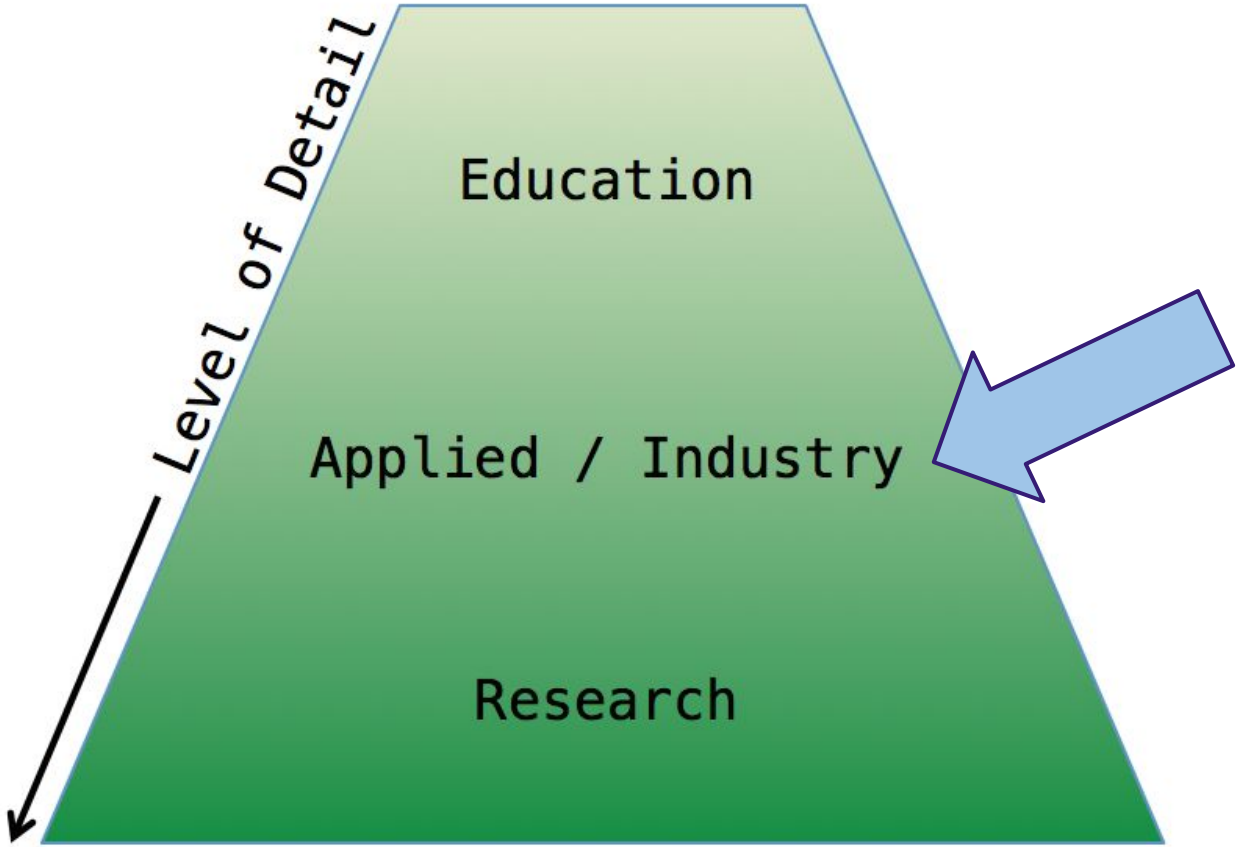


Testing!

```
Ran 366 tests in 136.939s

OK
```

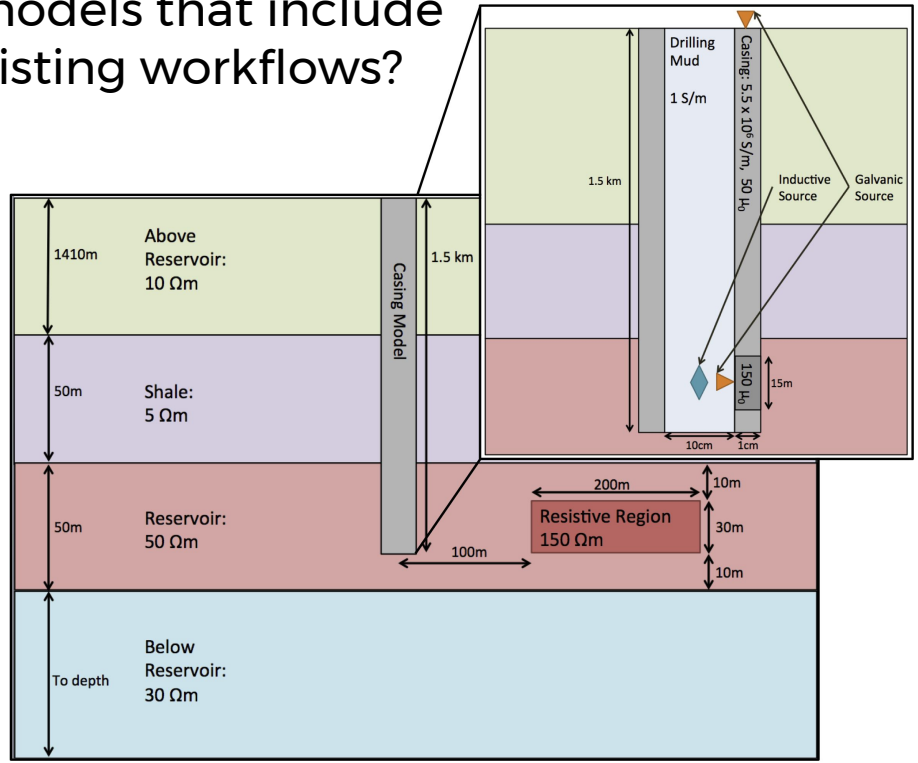
Base Class Inheritance

```
class BaseEMProblem(Problem.BaseProblem):
class BaseFDEMProblem(BaseEMProblem):
class ProblemFDEM_j(BaseFDEMProblem):
```


Level of Detail
Education
Applied / Industry
Research

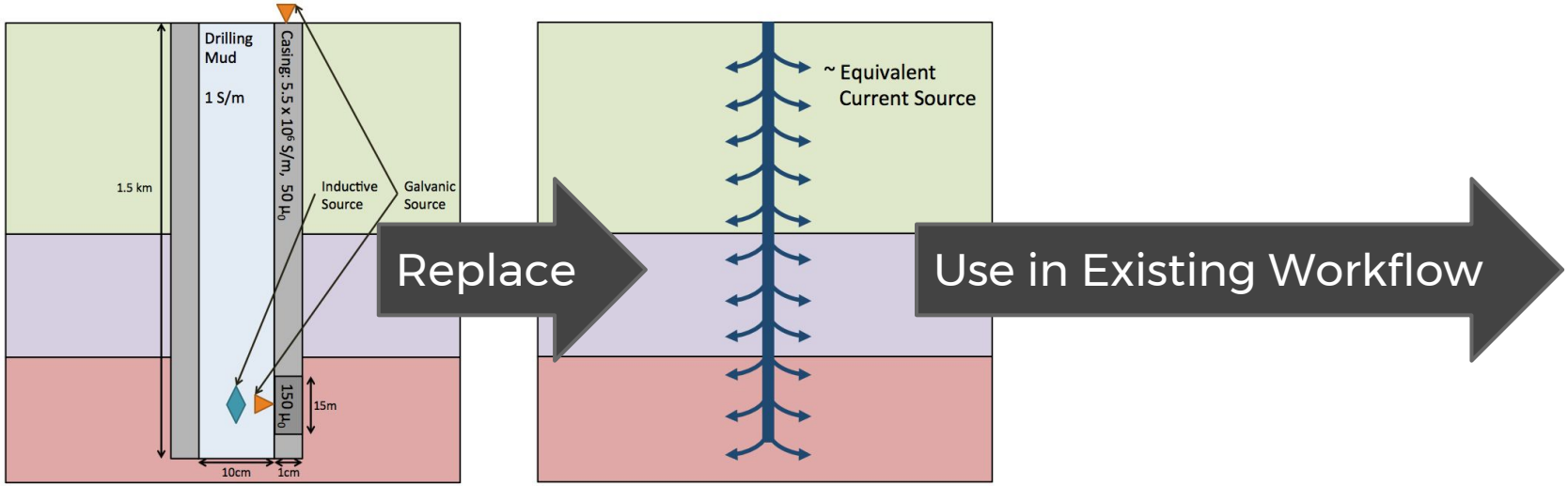
Applied / Industry

Problem: How do we plug in electromagnetic models that include well casing into existing workflows?



Applied / Industry

Approach: Replace complex casing model with a simpler model that can be included in existing codes



Applied / Industry

Resources & Practices:

Testing!

```
Ran 366 tests in 136.939s

OK
```

Packaging

```
>>> computeCasingCurrents(geologyModel,casingModel,sourceType,sourceLoc)
```

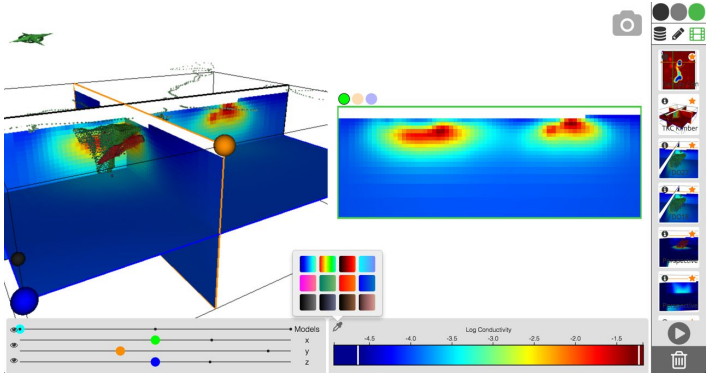
Versioning

Latest release

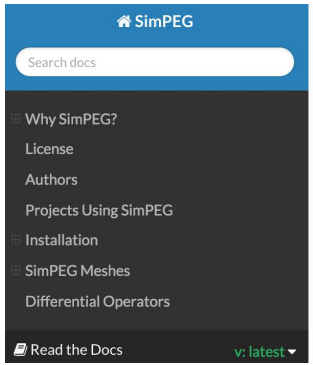
v0.1.3

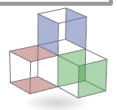
0c432e1

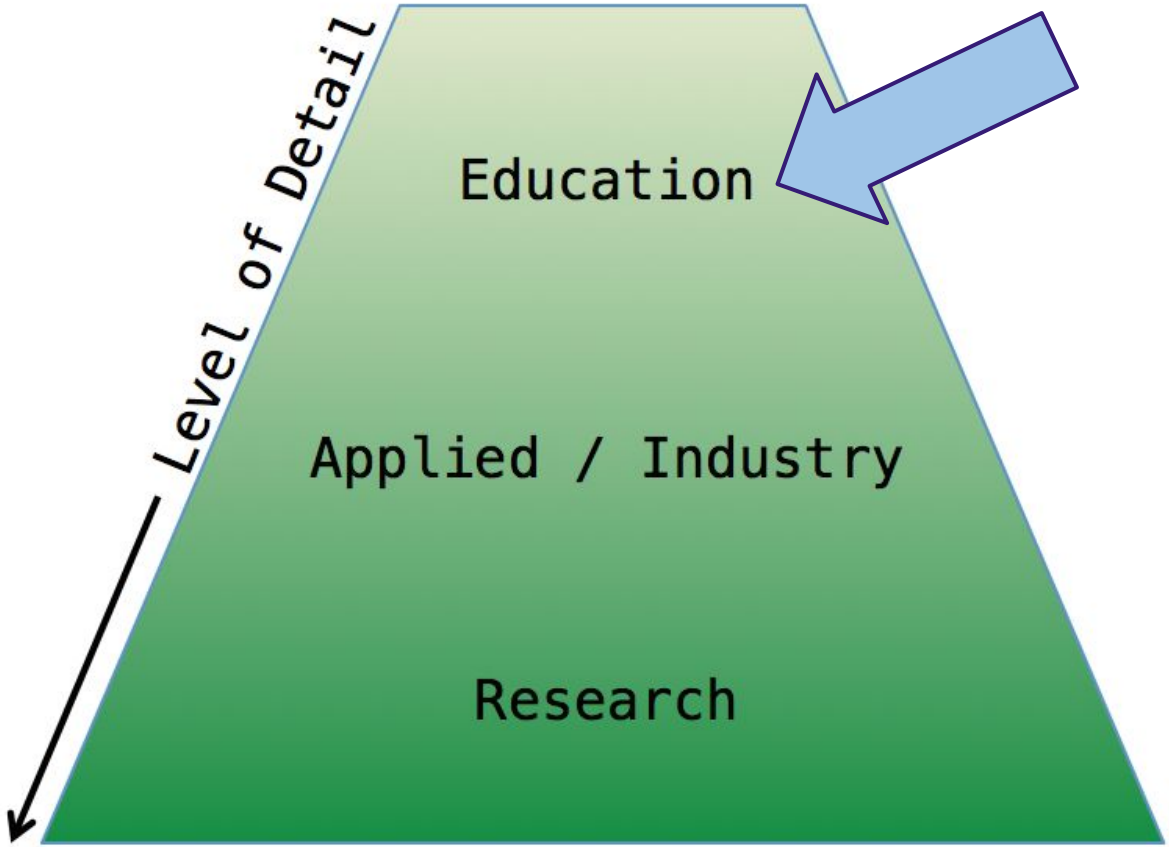
Visualization



Documentation



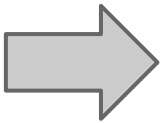


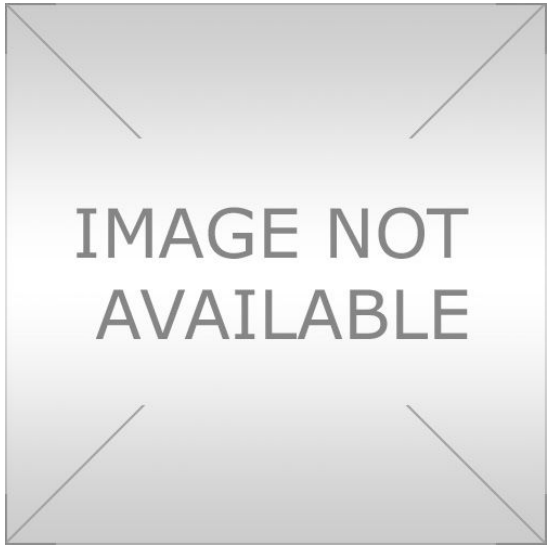

Level of Detail
Education
Applied / Industry
Research

Education

Problem: Electromagnetics??

$$\nabla \times \vec{E} + i\omega \vec{B} = 0$$
$$\nabla \times \mu^{-1} \vec{B} - \sigma \vec{E} = \vec{J}_s$$





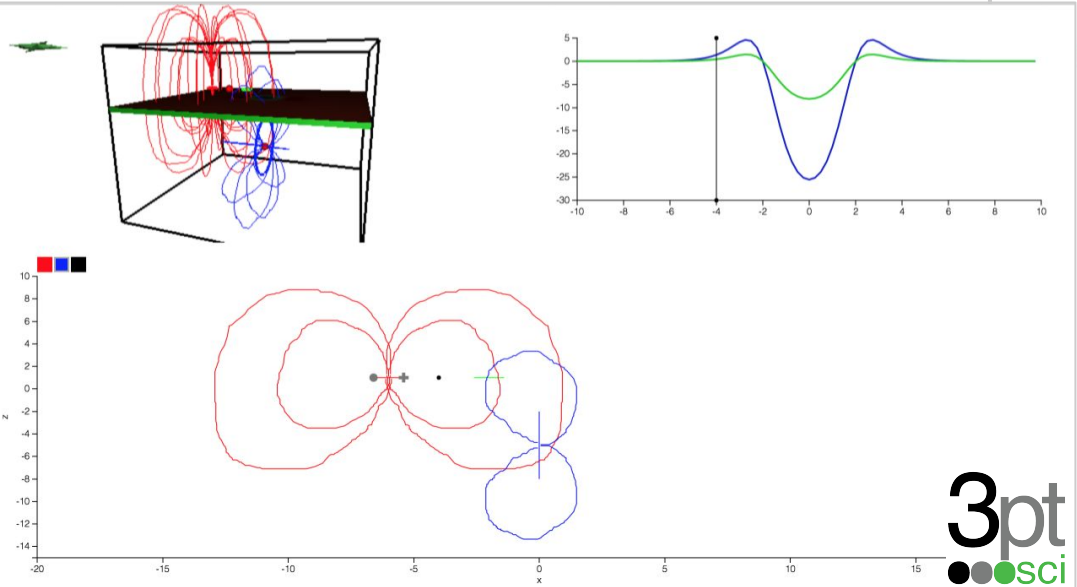
Education

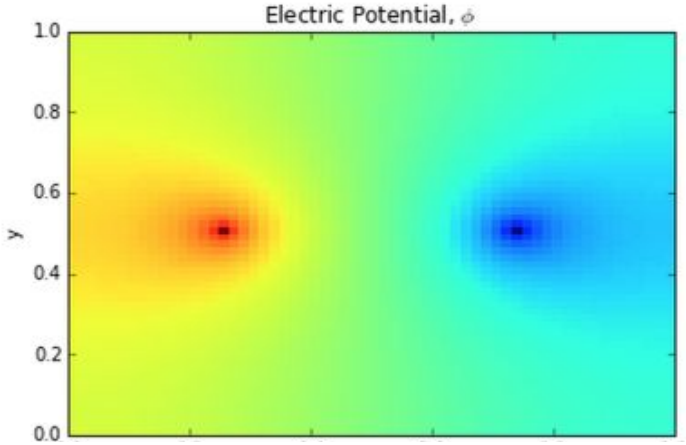
Approach: Lower barriers to entry by exposing an appropriate level of detail and making it interactive!

DC Resistivity

$$\nabla \cdot \sigma \nabla \phi = -s$$

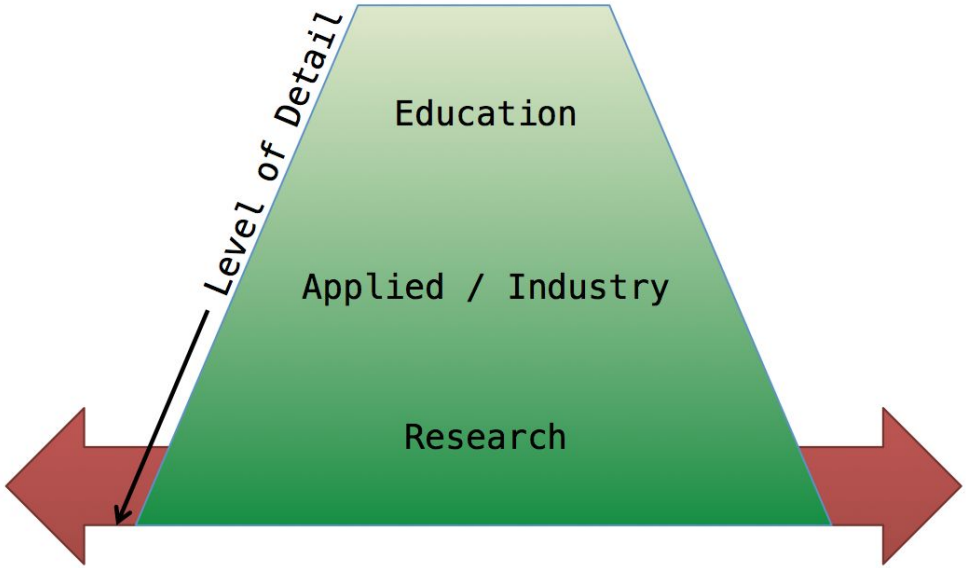
```
In [9]: # Construct A Matrix
        Div, Sigma, Grad = getOperators(mesh,sigma)
        A = Div * Sigma * Grad # looks like the equation!
        Ainv = Solver(A)

In [10]: phi = Ainv * -s

In [11]: mesh.plotImage(phi)
         plt.title('Electric Potential, $\phi$')

11]: <matplotlib.text.Text at 0x108b6e210>
```





https://github.com/simpeg/simpegExamples

Where we are headed



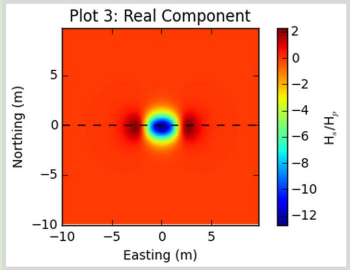
Package	State
SimPEG	✔
simpegEM	🔄
simpegMT	🔄
simpegFLOW	🧪
simpegDC	🧪
simpegPF	🧪
simpegSEIS	🔧
simpegGPR	🔧

Education

param 1

param 2

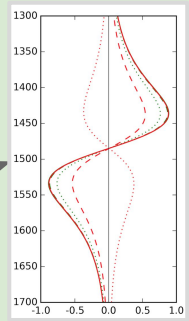
System



Industry & Applied

```
>>> import SimPEG
>>> SimPEG.solveMyProblem(someParameters)
```

Box that solvesMyProblem



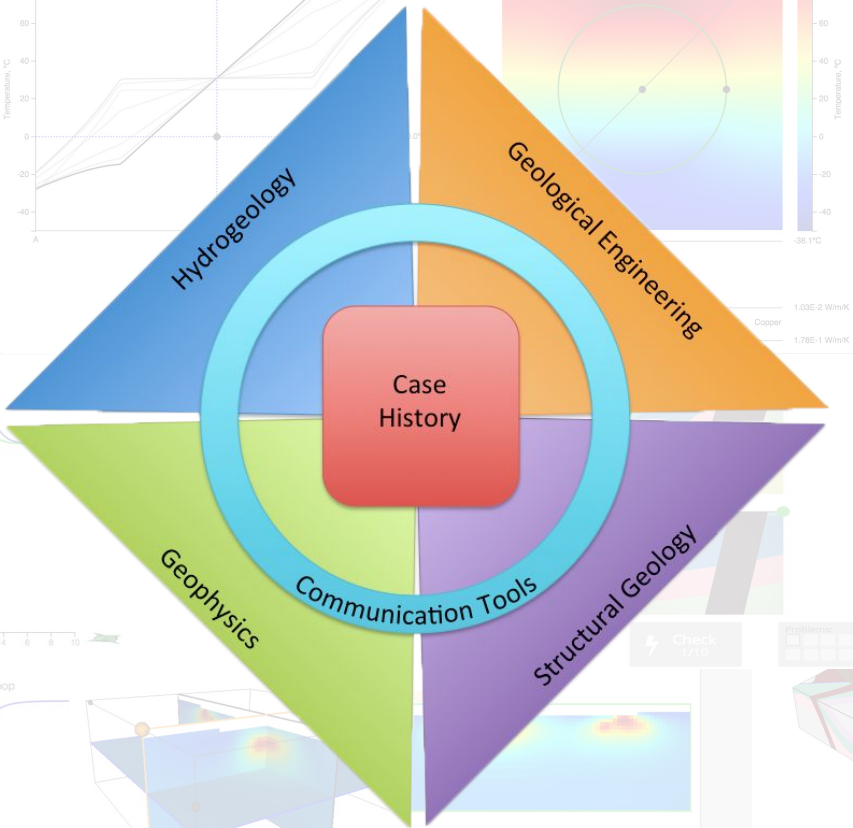
Research

Question

Box with building blocks

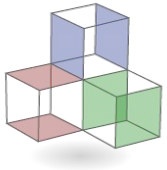
Methodology

GeoSci.xyz



... coming soon

Want more?



simpeg.xyz

github.com/simpeg

3pt.xyz

lindsey@simpeg.xyz