Developing Domain-Specific Languages for Ocean Modeling

EMLS'21 – Project OceanDSL

Reiner Jung, Sven Gundlach,
Serafim Simonov, Wilhelm Hasselbring

Kiel-University

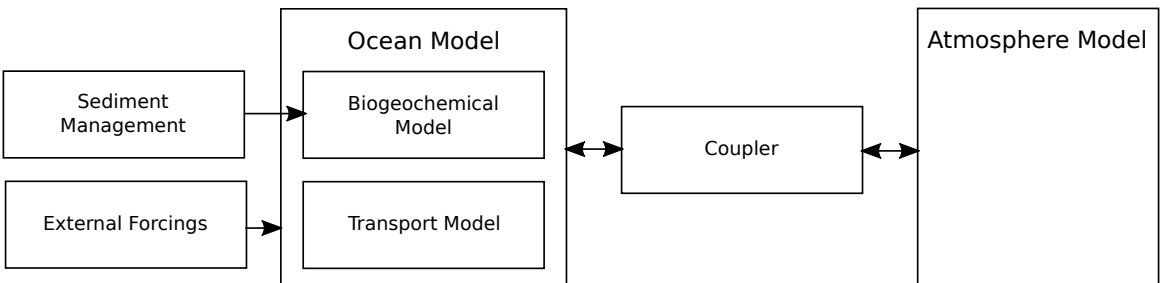
23rd February 2021

OceanDSL

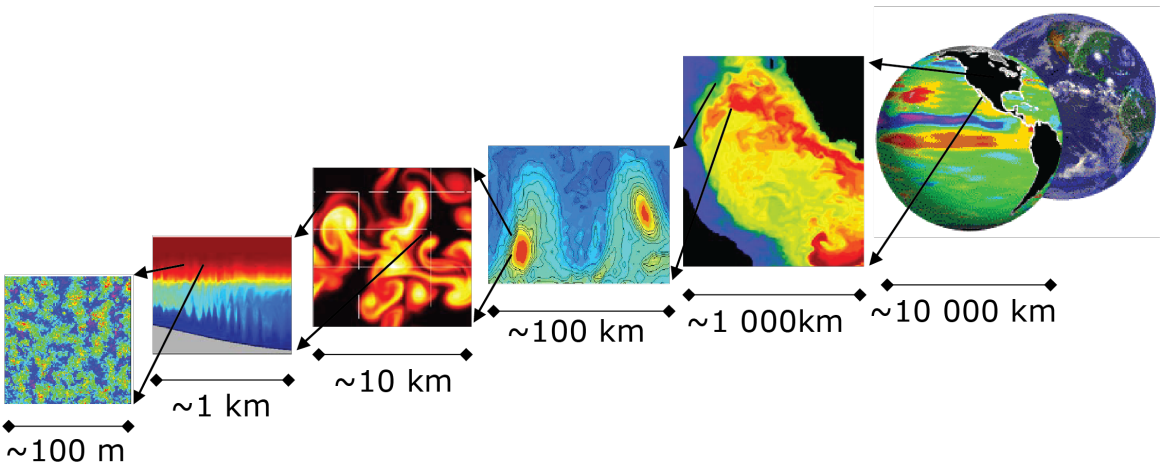
KIEL MARINE SCIENCE
CONNECTED RESEARCH

- Project Goal: DSLs to support Ocean Modeling
- Domain analysis: Thematic Analysis [Braun and Clarke 2006]
- Example: Configuration and Parameterization DSL

What is Ocean Modeling?



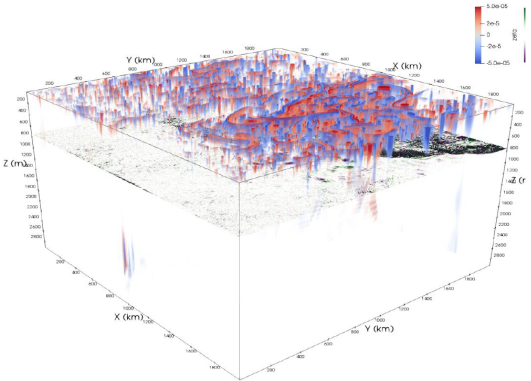
Different Scales in Modeling

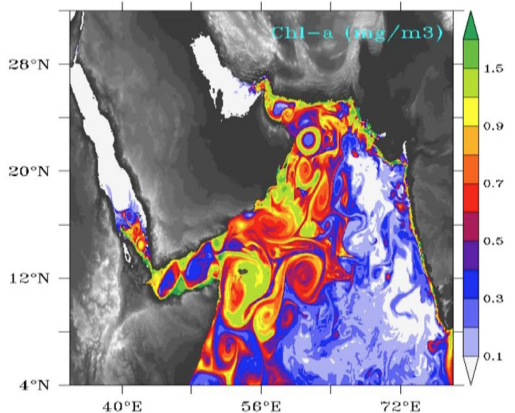


[MITgcm Project 2020]

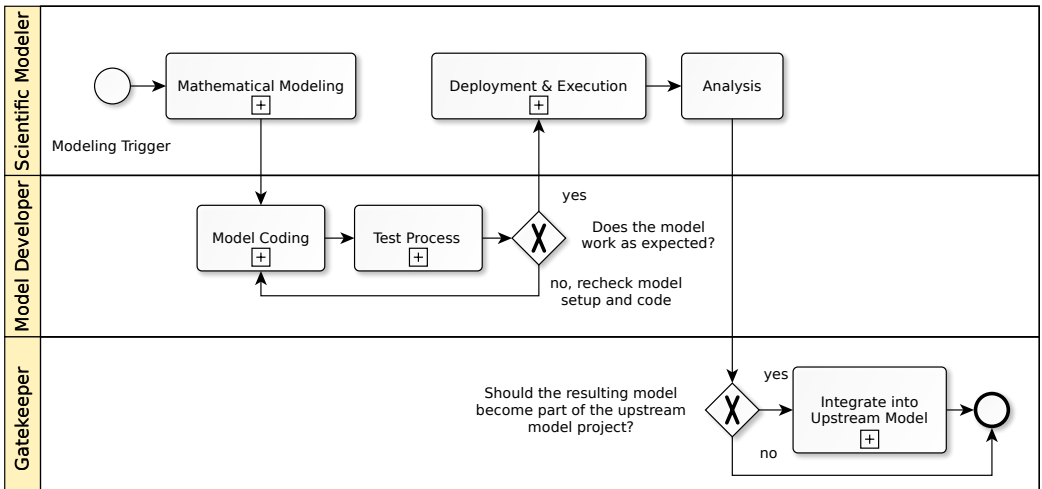
Fluxes of Heat, Carbon and Oxygen at SWOT Scales

[Smith and Abernathey 2017]





Simplified Ocean Modeling Process



Domain Characteristics

CAU

Models

- Long-living systems
- Implemented in Fortran 77, 90, C++ and Python
- Feature management by `#ifdef`

Editors

- Vi, Vim, Emacs and Xcode
- In general no IDEs
  (except Emacs, and PyCharm in rare cases)

Build system

- make, cmake, shell scripts, perl

DSLs in Ocean Modeling

- External DSLs, e.g., Dusk/Dawn MeteoSwiss [MeteoSwiss 2020]
- Embedded DSLs, e.g., Psyclone [Adams et al. 2019]

Views & Aspects

- Transport Model Specification
- Bio-geo-chemical Modeling
- Configuration and Parameterization
- Deployment

Configuration Header

```
include size

barotropic_gyre : mitgcm
```

```
features ALLOW_FRICTIONHEATING

parameters
PARM01:
   viscAh = 4.E2
   f0 = 1.E-4
   beta = 1.E-11
   rhoConst = 1000.0
   gBaro = 9.81
```

Modules

```
module cost:
   features ALLOW_EGM96_ERROR_COV
   cost_nml:
      mult_atl = 0.
      mult_test = 0.
```

```
diagnostics:
   diagMdsDir = "some-dir"
   format = netcdf
   diagSt_regMaskFile = "regMask_lat24.bin"
   set_regMask(1:3) = [ 1, 1, 1 ]
   val_regMask(1:3) = [ 1., 2., 3. ]
   "first-out.log":
      logmode = snap
      frequency = 10
      missing_value = 5.0
      fields(1:2) = [ SDIAG1, SDIAG2 ]
      levels(1:2) = [ 1, 2 ]
```

Summary

- Introduced the domain of **ocean modeling**
  - Main process
  - Domain properties
- Presented the **configuration and parameterization DSL**

Questions

C|A|U

Language related aspects

- Which syntactical style should we use?
  - YAML, CPP, C, Python
  - Familarity might be relevant
  - Structures must be as clear and simple
- How could we address modularization of the configuration?
  - Include, override, interfaces, immutuals

Technical and social aspects

- How to introduce DSLs into the domain?
  - Are there methods from other domains which we could use here?
  - What are usual methods and arguments to hinder the introduction of DSL?
  - How can we address them?
- How to organize maintenance after the project ends?
  - How to motivate institutions to commit themselves?
  - How to minimize maintenance?

Bibliography I

Adams, S.V. et al. (2019). "LFRic: Meeting the challenges of scalability and performance portability in Weather and Climate models." In: *Journal of Parallel and Distributed Computing* 132, pp. 383–396. DOI: 10.1016/j.jpdc.2019.02.007.

Braun, Virginia and Victoria Clarke (2006). "Using thematic analysis in psychology." In: *Qualitative Research in Psychology* 3.2, pp. 77–101. DOI: 10.1191/1478088706qp063oa.

MeteoSwiss (2020). *Dawn – Compiler toolchain to enable generation of high-level DSLs for geophysical fluid dynamics models*. URL: https://github.com/MeteoSwiss-APN/dawn.

MITgcm Project (2020). *MITgcm user manaual*. URL: https://mitgcm.readthedocs.io/en/latest/overview/overview.html.

Smith, Shafer and Ryan Abernathey (2017). *Fluxes of Heat, Carbon and Oxygen at SWOT Scales*. URL: https://swot.jpl.nasa.gov/documents/1521/.