Using MOM6

Alistair Adcroft

GFDL MOM6 team includes Raphael Dussin, Robert Hallberg, Stephen Griffies, Matthew Harrison, Hae-Cheol Kim, John Krasting, Marshall Ward, Niki Zadeh

Other contributors for presentation: Kate Hedstrom, Andrew Shao
“Using MOM6”
   - “Where to start?”

Every application is unique
   - Sometimes requires custom code specific to that one configuration

Building new configurations
   - Best to start from working example

1. Getting code/compiling/running
2. Controlling
   - Parameters
   - Diagnostics
3. Examples
   - Places to start
4. Model/repository structure
“Getting started” (Cloning, compiling, running)

• “MOM6 wiki”
  – (first result on google)

• Instructions for
  – Cloning (obtaining code)
  – Compiling (fairly portable)
  – Running (fairly standard)

• User-contributed

• Instructions assume some familiarity with linux & models
• Best limited to stand-alone ocean-only configurations
  – Coupled models are so much more complicated
• Does not cover working within an environment like GFDL’s FRE or CESM’s CIME
  – just low-level basics
Model input: run-time parameters

- Parameter syntax is key, value pairs
  
  \[ KH = 25. \]

  - Self-documenting runs

- Simple API

  - MOM6 always writes out
    - MOM_parameter_doc.all (everything)
    - MOM_parameter_doc.short (non-defaults)

- Bootstrapped parameter parser
  - namelist in input.nml

    ```
    &MOM_input_nml
    output_directory = '.',
    input_filename = 'n'
    restart_input_dir = 'INPUT',
    restart_output_dir = 'RESTART',
    parameter_filename = 'MOM_input',
    'MOM_saltrestore',
    'MOM_override'
    ```

- Typical setup
  - Baseline uses blank MOM_override
  - Perturbation runs concisely contained in MOM_override

- Lots of error checking

Using MOM6
• Reported in MOM\_parameter\_doc.layout

- Bitwise reproduces across layout
- Optimal tile size ~ 12x12-30x30
  - Tile size = NIGLOBAL/NIPROC,NJGLOBAL/NJPROC
- Halos ~3-4 most typically needed
  - Parameters NIHALO,NJHALO
  - Without high-order advection and certain choices of time-stepping
- Tile dimensions should be >= NIHALO,NJHALO
- Tiles may not be uniform!
  - Different sized tiles are allowed
• Many runs take longer than a single job submission
• Bitwise reproducibility across a restart boundary
• Online time-averaged diagnostics are not included in restarts
  – handled by FMS framework
• Input data usually read from INPUT/
• Diagnostic output is in current directory
• Restart files generally written to RESTART/
• Restart files are read from INPUT/
Controlling diagnostics

- FMS `diag_manager` parses `diag_table`

File definition

<table>
<thead>
<tr>
<th>File Name</th>
<th>Sections</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>ocean_daily</td>
<td>1, days, 1, days, time</td>
<td></td>
</tr>
<tr>
<td>ocean_month_snap</td>
<td>1, months, 1, days, time</td>
<td></td>
</tr>
<tr>
<td>ocean_month</td>
<td>1, months, 1, days, time</td>
<td></td>
</tr>
<tr>
<td>ocean_month_z</td>
<td>1, months, 1, days, time</td>
<td></td>
</tr>
<tr>
<td>ocean_annual</td>
<td>12, months, 1, days, time</td>
<td></td>
</tr>
<tr>
<td>ocean_annual_z</td>
<td>12, months, 1, days, time</td>
<td></td>
</tr>
<tr>
<td>ocean_scalar_month</td>
<td>1, months, 1, days, time</td>
<td></td>
</tr>
<tr>
<td>ocean_scalar_annual</td>
<td>12, months, 1, days, time</td>
<td></td>
</tr>
<tr>
<td>ocean_static</td>
<td>-1, months, 1, days, time</td>
<td></td>
</tr>
</tbody>
</table>

- Variable lists per file/module

- **MOM6** wraps `diag_manager`
  - **Registers** same diagnostic in multiple vertical coordinates, multiple names, xy-averages

- Available diagnostics written by MOM6 at run-time
  - `available_diags.0000`

- **Regional** diagnostics
• **double_gyre**
  – Wind driven gyre using stacked shallow water equations

• **Phillips_2layer**
  – Idealized channel model

• **flow_downslope**
  – Adjustment problem over topography using different coordinates

• **OM_05**
  – 1/2° ice-ocean global model
• jupyter notebook
  – (sorry about rainbow colormap)
  – This one uses matplotlib and scipy
• Much more needs to be added
  – Notebook for other examples
  – Others use netCDF4 instead of scipy
  – Will add xarray, seaborn examples
• Very much NOT advocating for one analysis system/style
• 2d density current

• Notebook is a treatise on how plotting in the vertical can go wrong

  – Idea is to explain how the model stores data in the vertical

    ... and how to look at the vertical in native space
• Hallberg, 2013
• Idealized zonal channel
  – Customized forcing
• No jupyter notebooks yet
  – plots were done with Ferret
• ½° global ice-ocean model
  – Uses GFDL SIS2 sea-ice model and GFDL coupler
• Uses GFDL vertical physics
  – ePBL, JHL, ...
• Non-eddying (coarse resolution)
• Uses GM and neutral-diffusion parameterizations

Adcroft et al., 2019
Experiment oriented repositories

- Using a repository for experiment development
  - treating configurations like code
- No new tools (just git)
- Provides history of experiment design
  - Recoverable / reproducible

- Used in other workflows
  - e.g. Payu, ROMS
• Layered repositories using sub-modules
  – Regression results
    • Output from regression tests
      – Platform dependent
    • Records specific version of configurations
  – Configurations
    • Input files (parameters)
    • Records specific versions of source
      – Including URLs (for forks)
  – Source for MOM6, FMS, SIS2, ...
    • Pure source code (+ packages)
config_src/
  • Selectively compiled
     – NCAR coupled mode: nuopc_driver + dynamic
     – Stand-alone ocean model solo_driver + dynamic
  • Alternative version of same code e.g.
     dynamic or dynamic_symmetric

Driver layer

External packages

Model
External packages

- Not compiled in place
- Symbolic links to required source live under src/ and point to pkg/
  - External packages often contain more than source and not all source compiles!
- Each package is a git submodule
  - using specific commit hash
Using MOM6

Code tree: main model

- **MOM6/**
  - config_src/
    - coupled_driver/
    - dynamic/
    - dynamic_symmetric/
    - mct_driver/
    - nuopc_driver/
    - solo_driver/
  - docs/
  - pkg/
    - CVMix-src/
    - geoKdTree/
    - GSW-Fortran/
    - MOM6_DA_hooks/
  - src/
    - ALE/
    - core/
    - diagnostics/
    - equation_of_state/
    - framework/
    - ice_shelf/
    - initialization/
    - ocean_data_assim/
    - parameterizations/
      - lateral/
      - vertical/
    - tracer/
    - user/

- **src/**
  - **Driver layer**
  - **External packages**
  - **Model**

  - Code for solving equations of motion, tracers, diagnostics, etc.
  - Always compiled
  - No CPP macros except for MEMORY and GRID macros
    - the few existing exceptions will be removed one day
Using MOM6
MOM6/

├── .testing/

│

│

│

└── docs/

.docs/

• Continuous integration
  – Runs tests on Travis-CI
    (soon also GitHub Actions)
  – Can be used for development

docs/

• Source for documentation
  hosted at
  https://mom6.readthedocs.io
  – Under dev. by K. Hedstrom
Future topics

- Verification and validation of MOM6 contributions
  - Marshall Ward
- Equations and algorithms
  - Bob Hallberg
- Lagrangian remap method
  - Stephen Griffies
- Analysis and tools
  - Raphael Dussin
- Ocean data assimilation interfaces
  - Matthew Harrison