Evolving Plans for MOM6 Development

Robert Hallberg and Marshall Ward
(NOAA/GFDL)
with contributions from numerous members of the MOM6 Development Team

MOM6 is available via https://github.com/NOAA-GFDL/MOM6-examples

MOM6 Development Plans at GFDL

- Configurations for Key NOAA Applications
  - Hierarchy of global configurations (1/4°, 1/8°, 1/12°, …) for weather to climate timescales
  - Regional use of MOM6 for Climate Fishery Initiative
- Algorithmic Development for Key Applications
  - New parameterizations (Eddy CPT, wave-mixing, …)
  - Better cryosphere/ocean coupling (Embedded ice, …)
- Code Management for a Growing Community
  - Documentation *(Well done Kate Hedstrom!)*
  - Testing & self-enforcing code conventions
  - Improving performance

The Vertical Lagrangian Remap method

(The flavor of ALE in Hycom and MOM6)

Solve equations in 2 phases:

– a **Lagrangian** dynamic update (shallow water eqns.)

– Vertical remapping to an arbitrary (**Eulerian**?) coordinate

Momentum eqn.:

$$\frac{\partial \vec{u}}{\partial t} + \dot{s}\frac{\partial \vec{u}}{\partial s} + (f + \nabla_s \times \vec{u})\hat{k} \times \vec{u} = -\frac{1}{\rho}\nabla_s p - \nabla_s\left(\phi + \tfrac{1}{2}\|\vec{u}\|^2\right) + \frac{1}{\rho}\nabla \cdot \tilde{\tilde{\tau}}$$

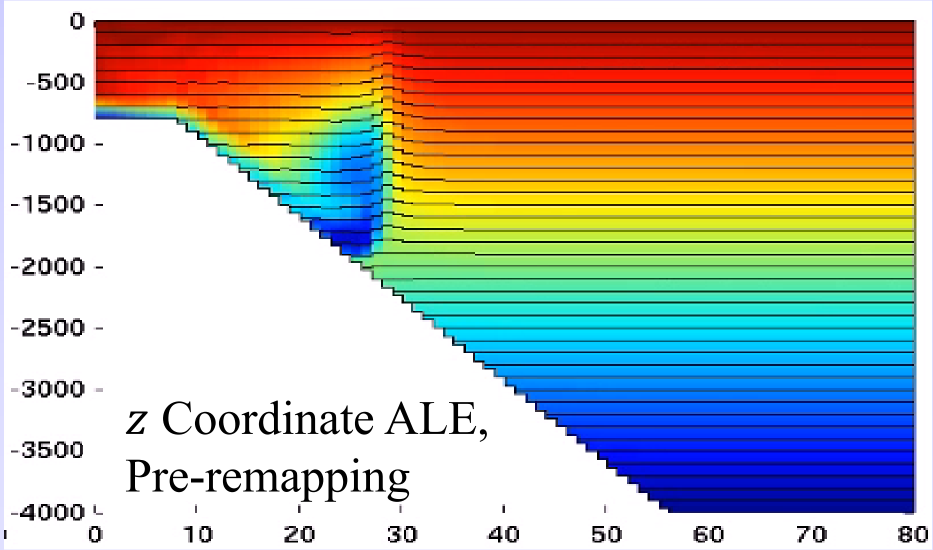
Continuity eqn.:

$$\left.\frac{\partial}{\partial t}\right|_s \left(\frac{\partial p}{\partial s}\right) + \nabla_s \cdot \left(\vec{u}\frac{\partial p}{\partial s}\right) + \frac{\partial}{\partial s}\left(\dot{s}\frac{\partial p}{\partial s}\right) = 0$$

Tracer eqn.:

$$\left.\frac{\partial}{\partial t}\right|_s \left(\frac{\partial p}{\partial s}\theta\right) + \nabla_s \cdot \left(\vec{u}\frac{\partial p}{\partial s}\theta\right) + \frac{\partial}{\partial s}\left(\dot{s}\frac{\partial p}{\partial s}\theta\right) = Q\frac{\partial p}{\partial s}$$

Dense-water Overflow Plume in Side-View

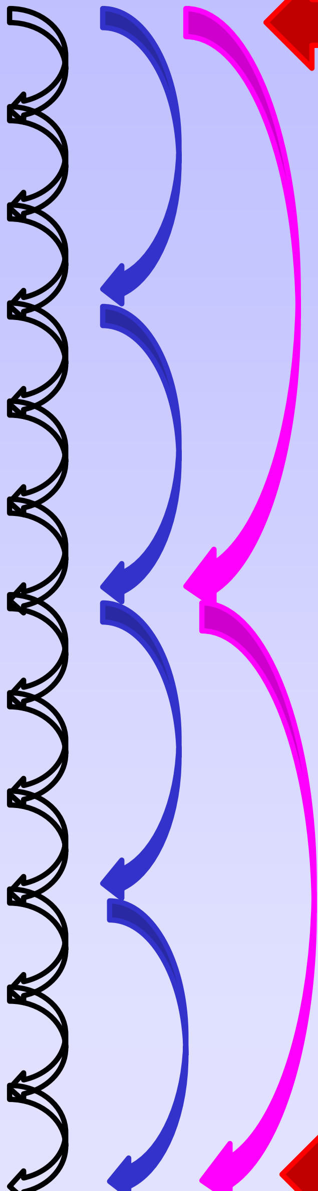


VLR advantages:

- Flexible vertical coordinates
- Remapping imposes no vertical CFL limit on timesteps
- Tracer advection not required to represent gravity waves

See Griffies, Adcroft and Hallberg (*JAMES*, 2020) for a detailed primer on VLR.

4 Time Stepping Cycles in MOM6



Barotropic time steps (2-d linear momentum, integrated continuity)

$$\frac{\partial \eta}{\partial t} + \nabla \cdot \left((D + \eta) \bar{u}_{BT} \right) = P - E \qquad \frac{\partial \bar{u}_{BT}}{\partial t} = -g \nabla \eta - f \hat{z} \times \bar{u}_{BT} + \bar{F}_{BT}$$

Lagrangian dynamics (3-d Stacked Shallow Water Eqns.)

$$\frac{\partial \vec{u}_k}{\partial t} + (f + \nabla_s \times \vec{u}_k) \hat{z} \times \vec{u}_k = -\frac{\nabla_s p_k}{\rho} - \nabla_s \left(\phi_k + \tfrac{1}{2} \|\vec{u}_k\|^2 \right) + \frac{\nabla \cdot \tilde{\tilde{\tau}}_k}{\rho}$$

$$\frac{\partial h_k}{\partial t} + \nabla_s \cdot (\vec{u} h_k) = 0$$

Tracer Advection, **Thermodynamics** and Mixing (Column physics)

$$\frac{\partial h_k}{\partial t} = (P - E)_k$$

$$\frac{\partial}{\partial t}(h_k \theta_k) + \nabla_s \cdot (\vec{u} h_k \theta_k) = Q_k^\theta h_k + \frac{1}{h_k} \Delta \left(\kappa \frac{\partial \theta}{\partial z} \right) + \frac{1}{h_k} \nabla_s (h_k K \nabla_s \theta)$$

Remapping and coordinate restoration

$$h_k^{new} = \Delta_k z_{Coord} \qquad \sum h_k^{new} = \sum h_k^{old}$$

$$\vec{u}_k^{new} = \frac{1}{h_k} \int_{z_{k+\frac{1}{2}}}^{z_{k+\frac{1}{2}} + h_k} \vec{u}^{old}(z') dz' \qquad \theta_k^{new} = \frac{1}{h_k} \int_{z_{k+\frac{1}{2}}}^{z_{k+\frac{1}{2}} + h_k} \theta(z') dz'$$

(Proto-) Operational Use of MOM6 in NOAA

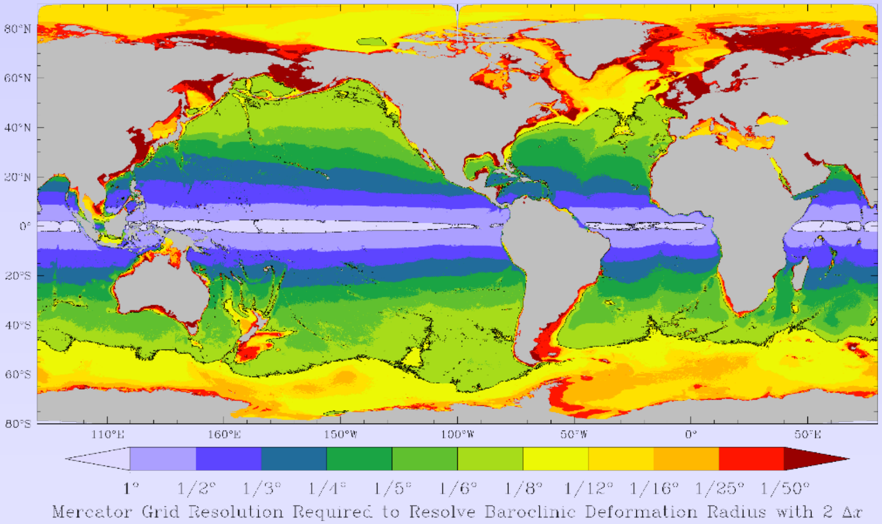
- CM4 (1/4°, 75L) & ESM4 (1/2°, 75L) global Coupled / Earth System Models (and variants)
- S2S forecasting, both at EMC and via the North American Multi Model Ensemble (NMME)
  - 1/4° global ocean in the prototype UFS-S2S-model
  - 1° global ocean in GFDL's "SPEAR" contribution
- 1/12° Global near-term forecast system at EMC
- Regional models for Climate-Fisheries Initiative
  - Tracer heavy configurations along U.S. coasts

Hierarchy of higher-resolution variants of CM4

- CM4 is the basis of a new series of higher resolution variants (1/4° => 1/8°, 1/12°, …)
- Climates have similarities, but expand eddy-permitting areas.
- Some versions will include interactive ice-shelves

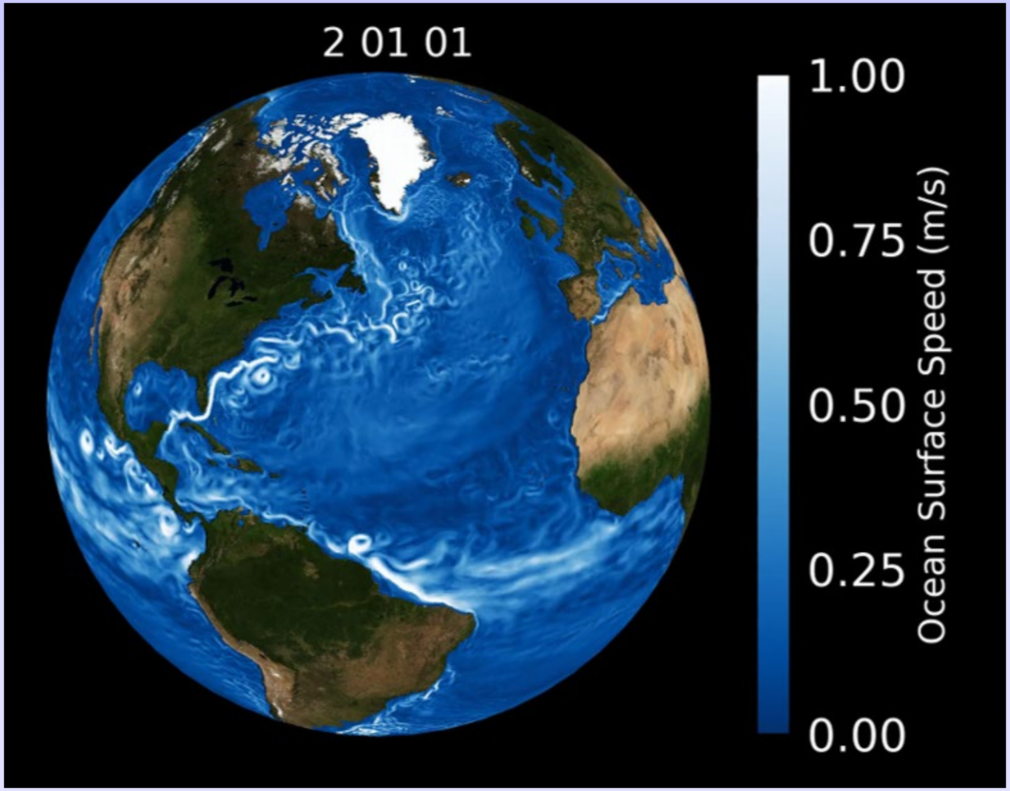
Resolution Required to Admit
1st Baroclinic Deformation Radius

$$L_{Def} = \sqrt{c_g^2 / \left(f^2 + 2\beta c_g\right)}$$



1° 1/4° 1/8° 1/16°

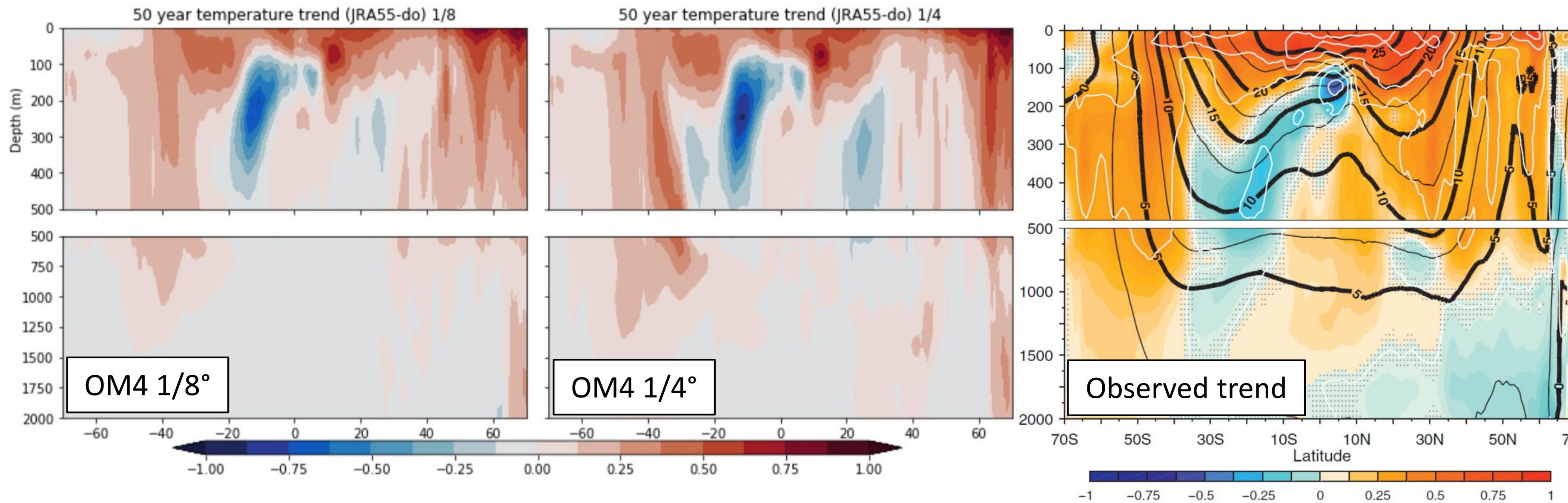
Ocean Surface Speed in 1/8° Variant of OM4



Animation courtesy Raf Dussin

Reanalysis Forced OM4 JRA55-do 1/8° & 1/4 °

50 year change of zonal average temperature [°C]

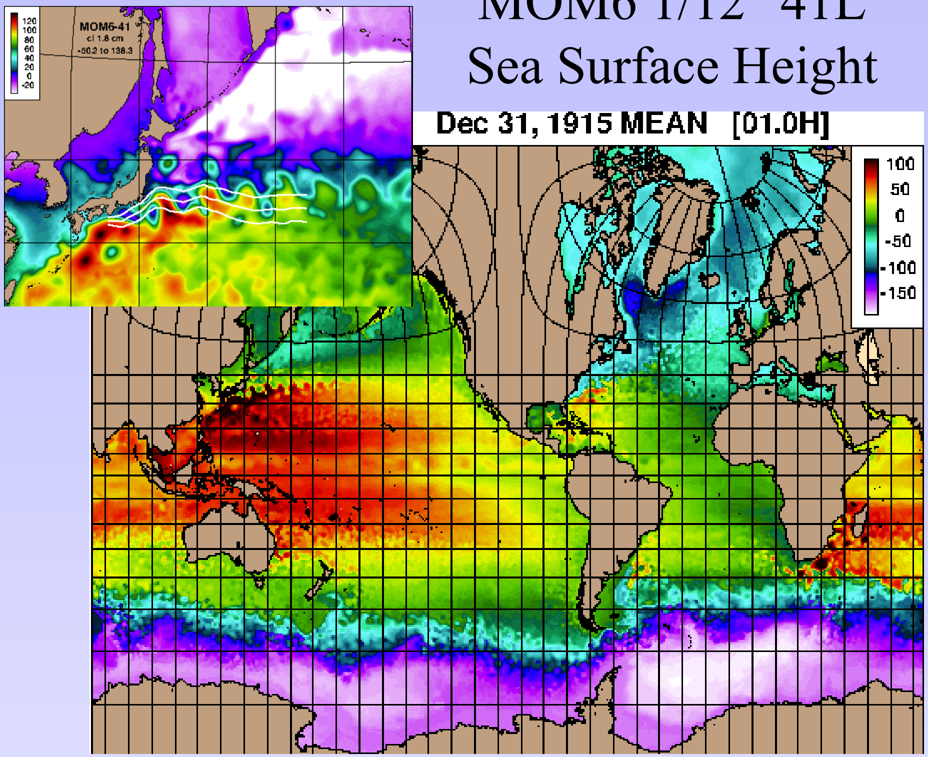


From Durack & Wiffels, 2010

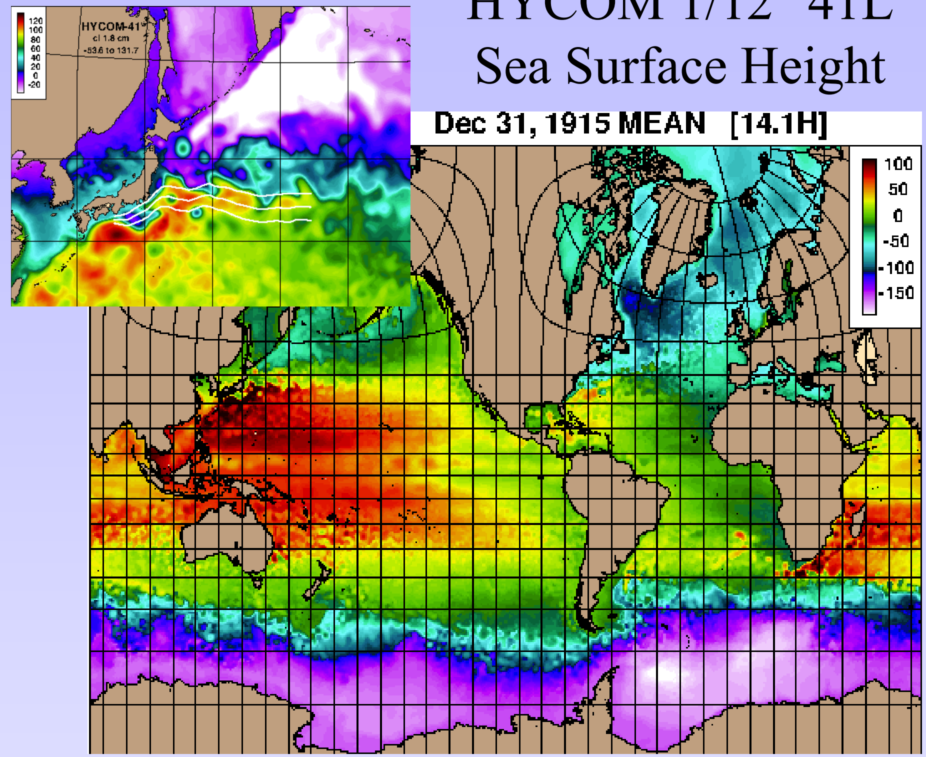
3rd cycle JRA55-do forcing
Historical atmospheric reanalysis by Japanese Meteorological Agency using 4d-Var atmospheric data assimilation

Unifying NOAA & Navy Operational Global Ocean Modeling Capabilities?

MOM6 1/12° 41L
Sea Surface Height



HYCOM 1/12° 41L
Sea Surface Height



Figures from Alan Wallcraft, FSU

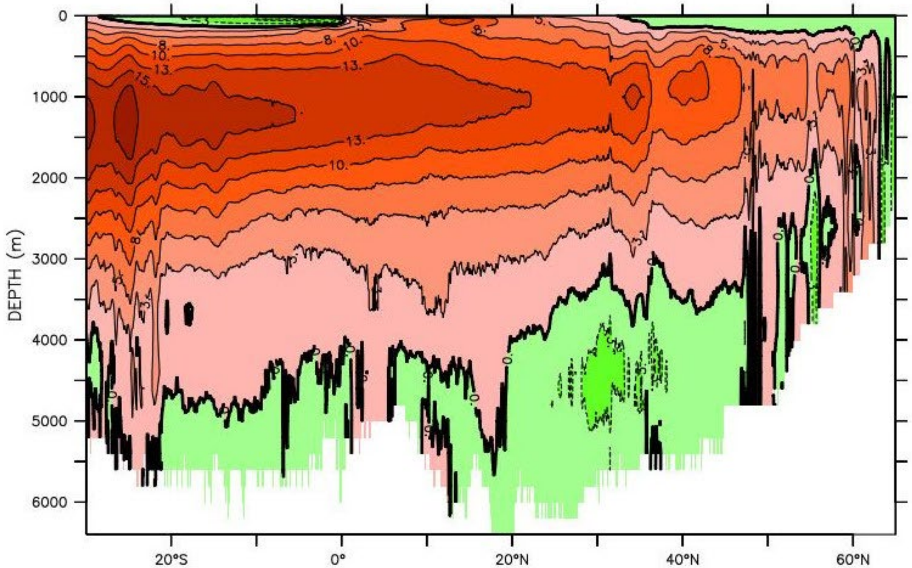
Compatible overall algorithmic designs, complementary strengths:

- Advanced climate capabilities (e.g., conservation, nonlinear EOS) with MOM6
- Navy has expertise in accurate near-term forecasts and extensive investments in real-time data ingest and assimilation system

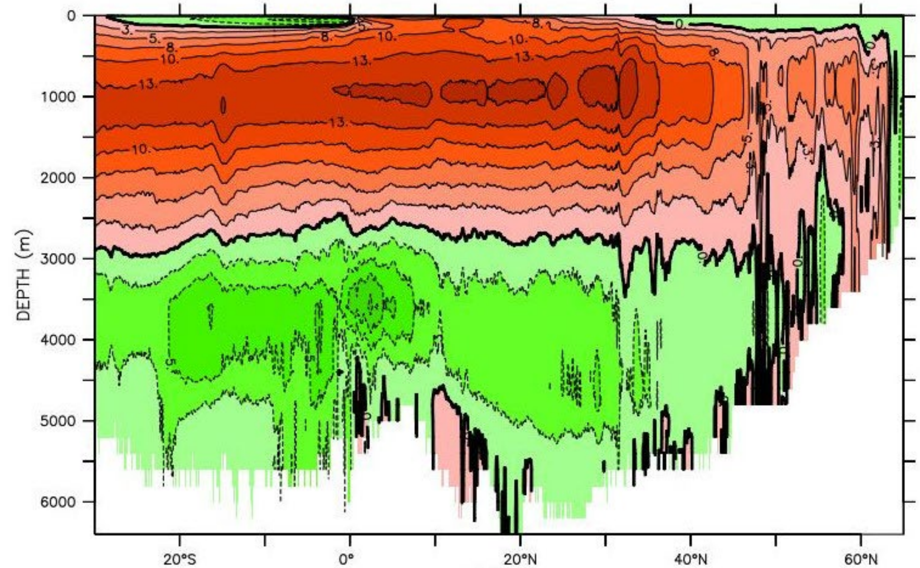
Comparisons are suggesting ideas for improving MOM6.

AMOC differences between MOM6 and HYCOM

Global 1/12 °, 41-Layer HYCOM



Global 1/12 °, 41-Layer MOM6

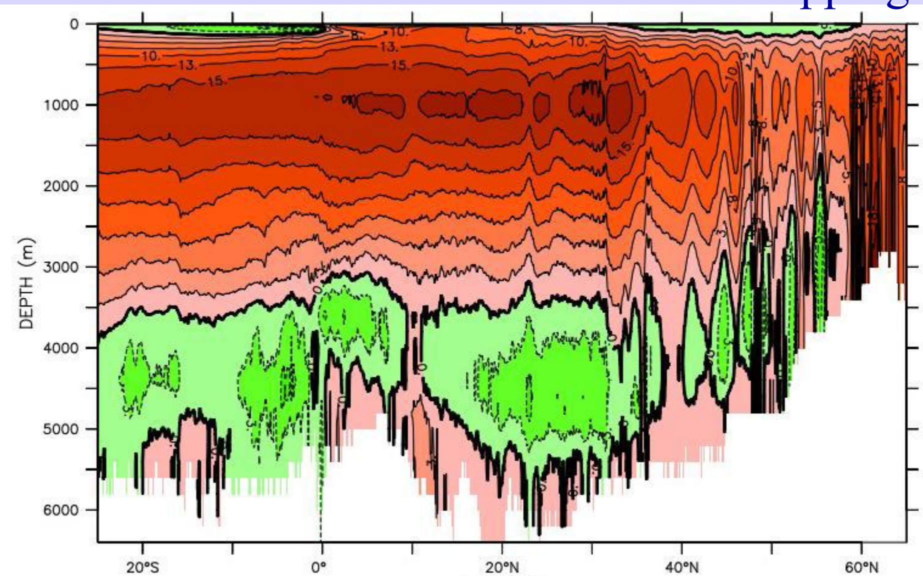


Similarly forced 1/12° global models are revealing some significant differences. Algorithms being scrutinized at GFDL due to this comparison include:

- Lateral drag near topography
- Vertical coordinate definition, and aggressiveness of remapping
- Bottom boundary layer turbulence params.
- Generation of topography (porous cells?)

Images Courtesy Alan Wallcraft

Atlantic 1/12 ° MOM6 w/ Slower Remapping

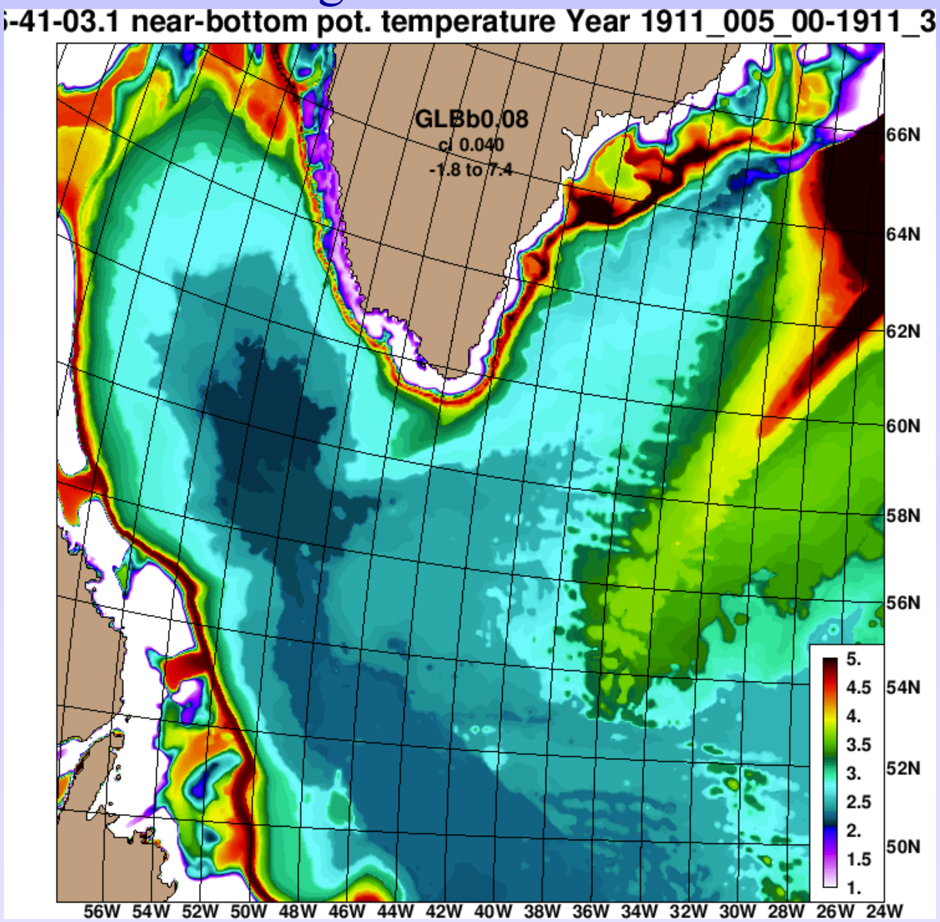


1/12° Near-bottom Temperature after 10 years

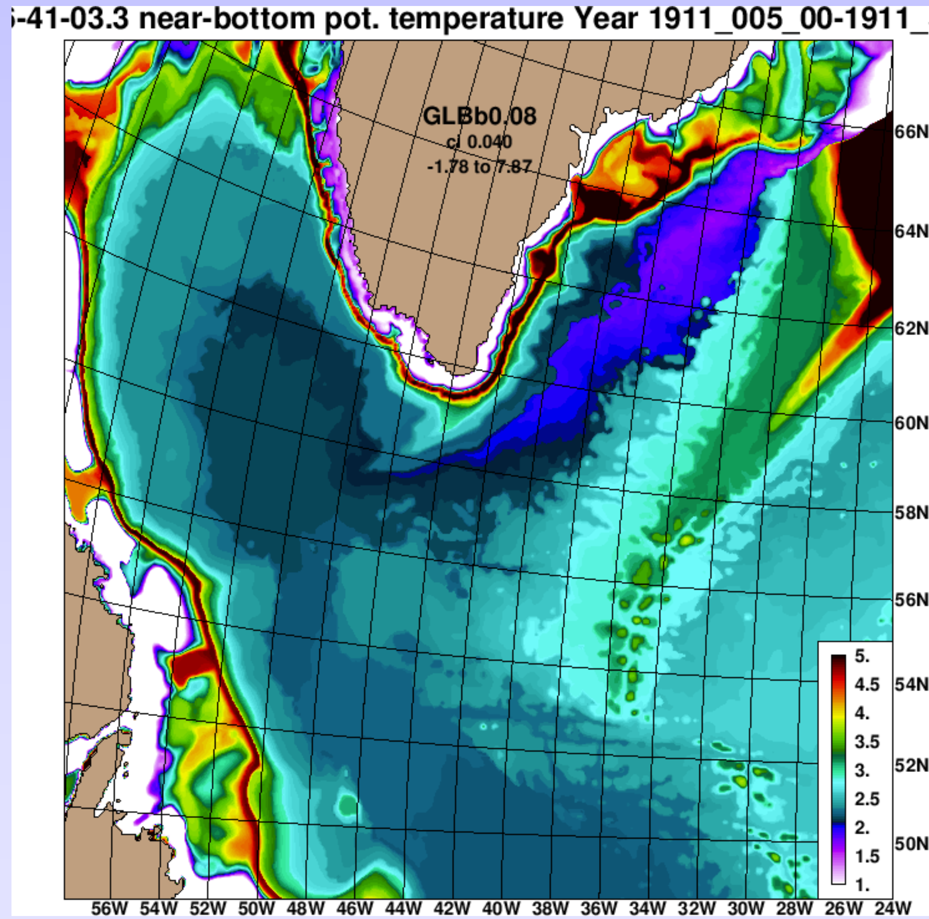
HYCOM/MOM6 comparison suggested parameter changes:

Less BBL mixing; KPP ~> ePBL; Slower remapping to coordinate

Original Parameters



Revised Parameters

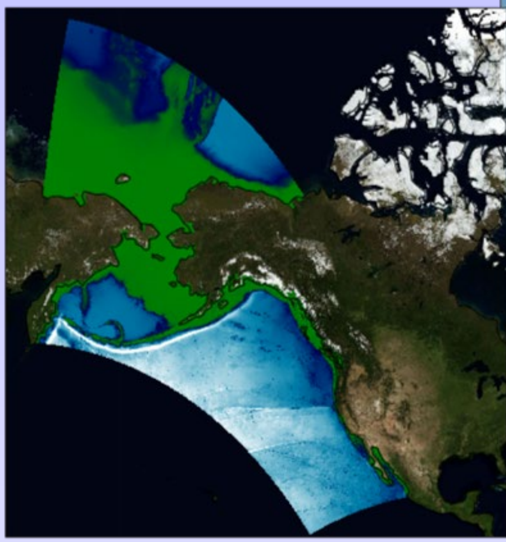


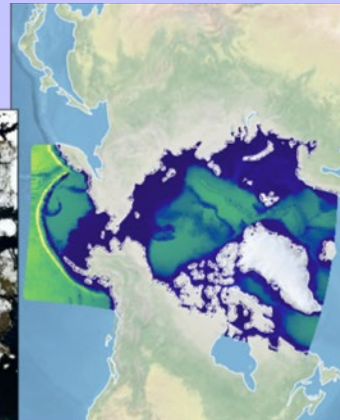
Figures from Hae-Cheol Kim, NOAA/EMC & GFDL

Using MOM6 for Regional Modeling

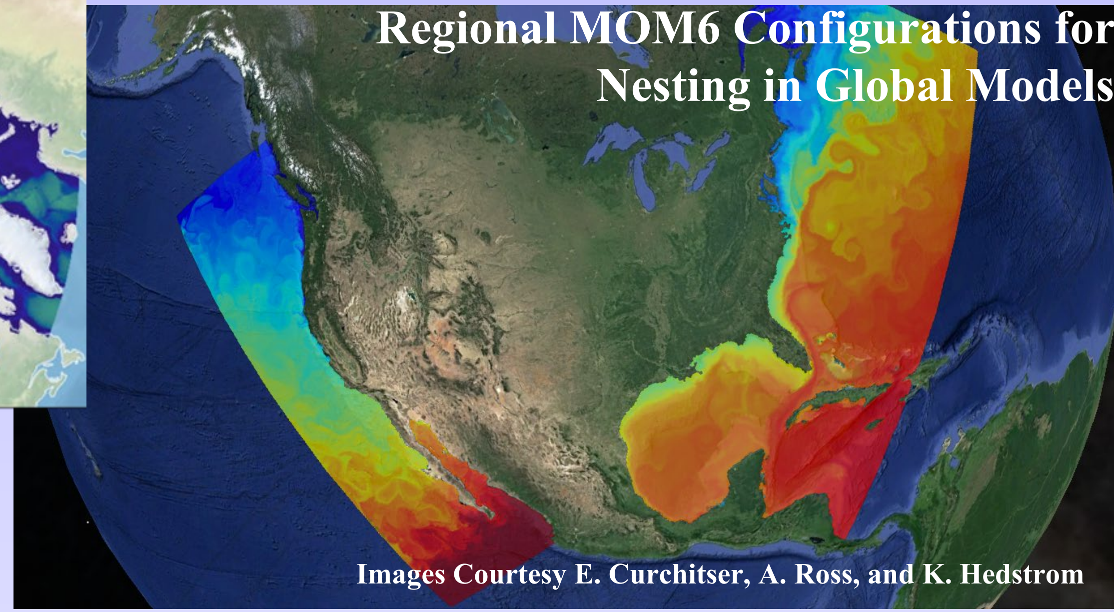
- Regional models permit much finer resolution than is practical in global configurations.

MOM6-NEP



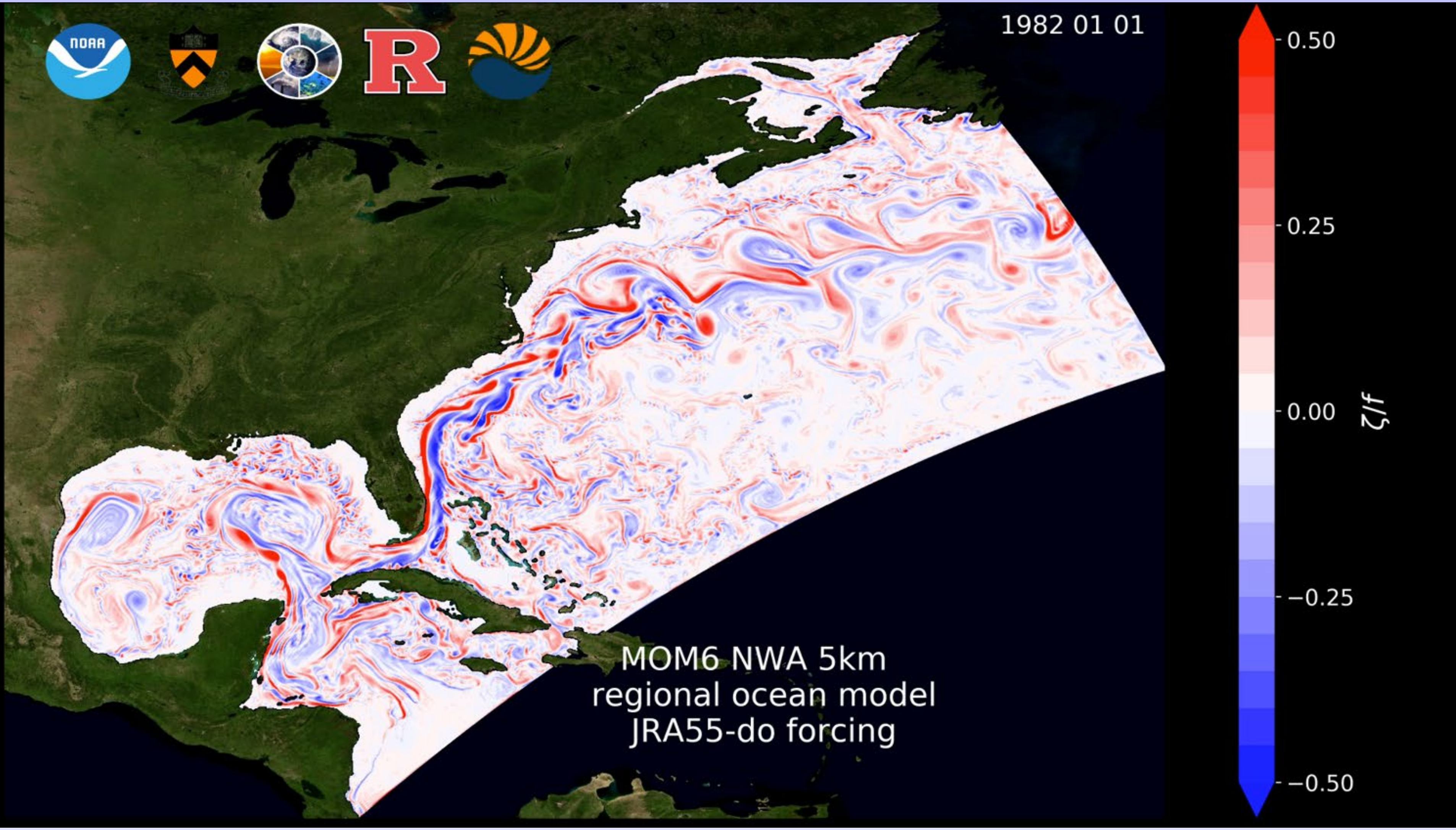


MOM6-Arctic

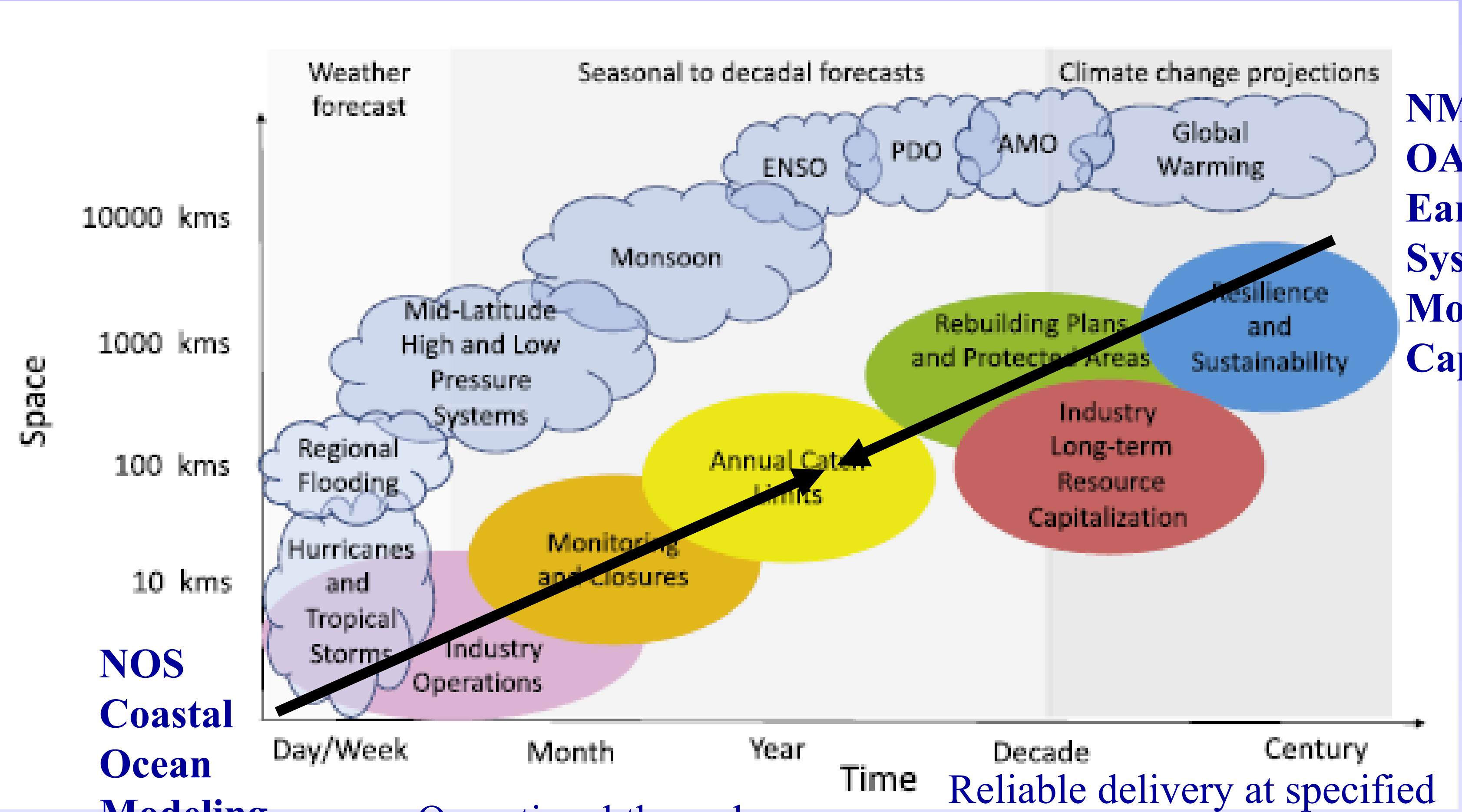


- The MOM6 algorithms offer advantages & efficiencies for Earth System Models with multiple bio-geo-chemical fields.
    - The same advantages apply to regional ecosystem models.
    - A regional modeling community has developed around MOM6 with NOAA-Fisheries Support and broader interest.

Using MOM6 for Regional Modeling



Fishery-related decisions over a broad range of spatial scales and time horizons



NMFS / OAR Earth System Modeling Capacity

NOS Coastal Ocean Modeling Program

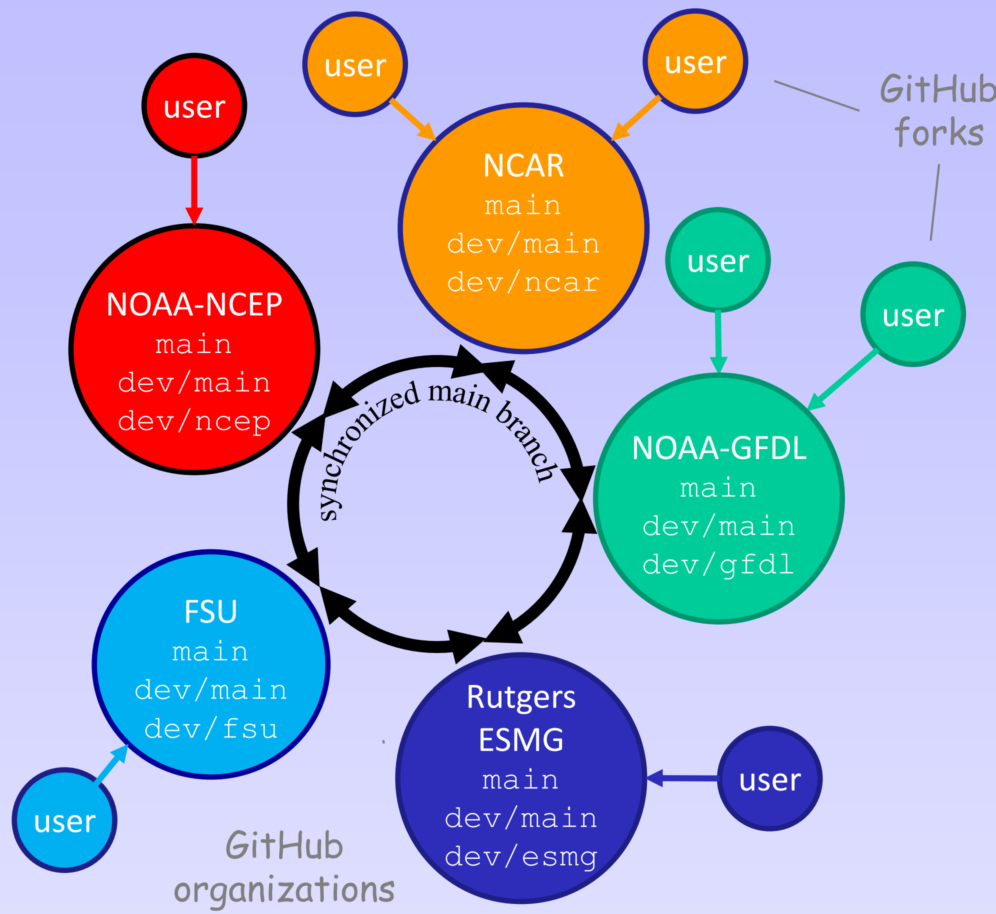
Operational through NOS/NWS partnerships

Reliable delivery at specified tolerances through OAR and regional team partnerships

Slide courtesy C. Stock

MOM6 Open Development via GitHub

- Developing MOM6 on **GitHub** has removed barriers to collaboration
  – Complete openness has attracted partners
- Continual + independent development
  – No "release delays"
- Numerous activities
  – 128 forks (as of Jan. '21)
  – 5 major hubs/partners ; 25 people have contributed > 1000 lines
- Synchronization to main branch occurs by ***consensus***
  – Hub-specific regression testing
  – Common code self-consistency testing



MOM6 Self-Consistency Tests

MOM6 has a series of self-consistency test which give bitwise identical answers:

- Parallelization processor count and layout
- Reproduction across restarts
- Index-space rotation (by 180°, 90° or 270°)
- Static or dynamic memory allocation
- Symmetric or non-symmetric memory
- Input parameter validation
- Dimensional consistency rescaling by 2^n

Failed self-consistency demonstrates the code is wrong.

Testing and Porting

The same robust testing that allows novice developers to contribute to MOM6 facilitates porting across computer architectures.

- Debugging often takes longer than writing or revising code in the first place.
- Detecting problems is the key to efficient debugging.
- The robust and definitive testing in MOM6 is good at detecting problems!
- Some MOM6 test cases can give identical answers across compilers and levels of optimization! (Transcendental functions are the issue for others.)

Machine-specific calls are encapsulated in a framework level that acts as the interface to the FMS infrastructure, facilitating porting.

- E.g., FMS2 will support separate I/O nodes. MOM6 code does not change.
- Localizes changes to adopt a different infrastructure (e.g., ESMF).

MOM6 already works with 4 interfaces to different couplers.

New code structure will work with various Infrastructure code

Detailed MOM6 performance analysis…

- Line profiling with perf
- Source + assembly
- Detailed resource usage
  - Wall time / cycles
  - Branching (if-else)
  - FLOPs
  - Memory usage
- Locate code “hot spots”

```
       │1263  Kv_tot(i,K) = Kv_tot(i,K) + (CS%Kvbbl-CS%Kv)*botfn
       │        vaddsd    -0x8(%r15,%r10,8),%xmm1,%xmm0
       │1264  h_shear = 0.5*(hvel(i,k) + hvel(i,k-1) + h_neglect)
       │        vmulsd    %xmm10,%xmm7,%xmm1
       │1263  Kv_tot(i,K) = Kv_tot(i,K) + (CS%Kvbbl-CS%Kv)*botfn
       │        vmovsd    %xmm0,-0x8(%r15,%r10,8)
       │1268  a_cpl(i,K) = Kv_tot(i,K) / (h_shear*GV%H_to_Z + I_amax*Kv_tot(i,K))
  0.02 │2c8f:   vmulsd    %xmm0,%xmm5,%xmm8
  0.24 │        vmulsd    %xmm1,%xmm2,%xmm1
  0.73 │        vaddsd    %xmm1,%xmm8,%xmm9
  6.40 │        vdivsd    %xmm9,%xmm0,%xmm0
 38.29 │        vmovsd    %xmm0,-0x8(%r9,%r10,8)
       │1248  do K=nz,2,-1 ; do i=is,ie ; if (do_i(i)) then
  0.02 │2ca7:   inc       %r10
       │        cmp       %rbx,%r10
  0.02 │        jb        2bbe
       │        mov       -0x1c8(%rbp),%rdx
  0.14 │        mov       -0x1b0(%rbp),%r8
       │        mov       -0x1c0(%rbp),%rbx
```

Slide Courtesy Marshall Ward

… and performance improvement

Loop vectorization

```
do j=Jsq,Jeq+1 ; do i=Isq,Ieq+1
  if ((CS%Smagorinsky_Kh) .or. (CS%Smagorinsky_Ah)) then
    Shear_mag = ...
  endif

  if (CS%better_bound_Ah .or. CS%better_bound_Kh) then
    hrat_min = ...
    visc_bound_rem = ...
  endif

  if ..
  if ..
  if ..

  ! > 200 lines of if-blocks
enddo; enddo
```

```
if ((CS%Smagorinsky_Kh) .or. (CS%Smagorinsky_Ah)) then
  do j=Jsq,Jeq+1 ; do i=Isq,Ieq+1
    Shear_mag(i,j) = ...
  enddo ; enddo
endif

if (CS%better_bound_Ah .or. CS%better_bound_Kh) then
  do j=Jsq,Jeq+1 ; do i=Isq,Ieq+1
    hrat_min(i,j) = ...
  enddo ; enddo

  if (CS%better_bound_Kh) then
    do j=Jsq,Jeq+1 ; do i=Isq,Ieq+1
      visc_bound_rem(i,j) = ...
    enddo ; enddo
  endif
endif
```

- If-blocks inside do loops impede vectorization and pipelining
- RHS was ~2x faster than LHS in horizontal_viscosity()
- Say “NO” to do – if – enddo, say “YES” to if – do – endif

Slide Courtesy Marshall Ward

Code analysis with flint

- Static code analysis
- Code style compliance
  - Parentheses (Op. Order)
  - Multiple divisions / statement
  - Code indentation
- Accurate tokenization of many scientific codes (MOM5/6, UM, CICE, …)
- Docstrings (as Doxygen)

```
S: subroutine update_surface_waves ( g , gv , us , day , dt , cs )
D: type ( wave_parameters_cs ) , pointer :: cs
D: type ( ocean_grid_type ) , intent ( inout ) :: g
D: type ( verticalgrid_type ) , intent ( in ) :: gv
D: type ( unit_scale_type ) , intent ( in ) :: us
D: type ( time_type ) , intent ( in ) :: day
D: type ( time_type ) , intent ( in ) :: dt
D: integer :: ii , jj , kk , b
D: type ( time_type ) :: day_center
E: day_center = day + dt / 2
C: if ( wavemethod == testprof ) then
e:  elseif ( wavemethod == surfbands ) then
C:  if ( datasource == dataovr ) then
e:   call surface_bands_by_data_override ( day_center , g , gv , us , cs )
e:   elseif ( datasource == coupler ) then
e:   elseif ( datasource == input ) then
C:   do b = 1 , numbands
C:    do ii = g % isdb , g % iedb
C:     do jj = g % jsd , g % jed
e:      cs % stkx0 ( ii , jj , b ) = cs % prescribedsurfstkx ( b )
C:      enddo
C:     enddo
C:    do ii = g % isd , g % ied
C:     do jj = g % jsdb , g % jedb
e:      cs % stky0 ( ii , jj , b ) = cs % prescribedsurfstky ( b )
C:      enddo
C:     enddo
C:    enddo
C:   endif
C:  endif
E: return
S: end subroutine update_surface_waves
```

Slide Courtesy Marshall Ward

MOM6 Ocean Model Discussion Points

The diverse and growing MOM6 community is building on existing strengths to develop advanced ocean modeling capabilities that respect the dynamics of the marine system across spatial and temporal scales.

MOM6 has demonstrated value for global- and regional-scale ocean forecasts, for weather- to climate-timescales (UFS and Regional ESMs to SPEAR and CM4), especially with marine ecosystem components.

"Open Development" and community agency have contributed to widespread adoption of MOM6 nationwide in the U.S. and around the world.

- Institutions have control over their own configurations and solutions, while benefitting from community model developments.
- Robust testing and quality control make Open Development work.
- Improving and maintaining code efficiency is an emerging challenge.

The Community Open Development of MOM6 is working…
but what can be done to make it work better?