Community Modeling

Critical aspects

Hendrik Tolman, NOAA / NWS / Office of Science and Technology Integration
Senior Advisor for Advanced Modeling Systems)

Hendrik.Tolman@NOAA.gov

Background

Key technical issues to enable efficient community modeling

- Infrastructure (previous talk)
- Repositories (tomorrow)
- Workflow (tomorrow)
- Coupled modeling (tomorrow)
- Governance (after this presentation)

Not a complete list

Not ordered

Key principles here, details elsewhere

Repositories I

How do you work together on code

Step 0: (ancient)

- Single code at center, no version control (as little as 10 years ago)

Step 1: (start of modern era)

- Code management through svn, internal server at NCEP
- Each collaborator needs to be “brought in” individually

Step 2a: (where we are now)

- Code management though Git, using Vlab as the platform
- More community tools
- More open, but still restrictive

Step 2b (soon)

- Add read only copy “outside” (Github)

Repositories II

How do you work together on code

Step 3: (target, 2019 ?)

- Full UFS lives on Github
  - All development (including EMC) on Github.
- “Gold Copy” within NOAA firewalls
  - VLab
  - Focus on part of UFS (intended) for operations
  - Source of code to be submitted to NCO
- NCO internal operational repository
  - No community impact
  - “flat code” available, but should not be considered

Definition of “Gold Copy” is from NOAA memo on use of Github and can be misleading.

Work Flow

Sub pieces in development

- JEDI, FV3GFS, CAM, MET+, (Obs. Proc., UPP, …)

Pieces identified in MoA

Also critical missing link still

CROW and CIME (NEMS and CMEP)

- Can we do a single workflow ?
  - NCO versus community requirements
- Work flow now focused on individual application
- Work flow for coupled models
- Develop component models with coupled model workflow


Thank You!