CSEG Update

Mariana Vertenstein
CESM Software Engineering Group

Outline

- CIME Update
- Data Assimilation in CESM
- New Efforts in Modularity
- Future Concurrency

CIME paradigm

CIME is being used to facilitate external collaborations

Infrastructure
PUBLIC Open Source Github Repository

Science code
Restricted or Public Repositories

Paradigm for DOE, NOAA, NSF Infrastructure Collaborations

Driver-Coupler
Data Models
Scripts
System/Unit testing
Mapping Utilities

CESM

DOE/ACME

ESPC and/or NOAA/NEMS

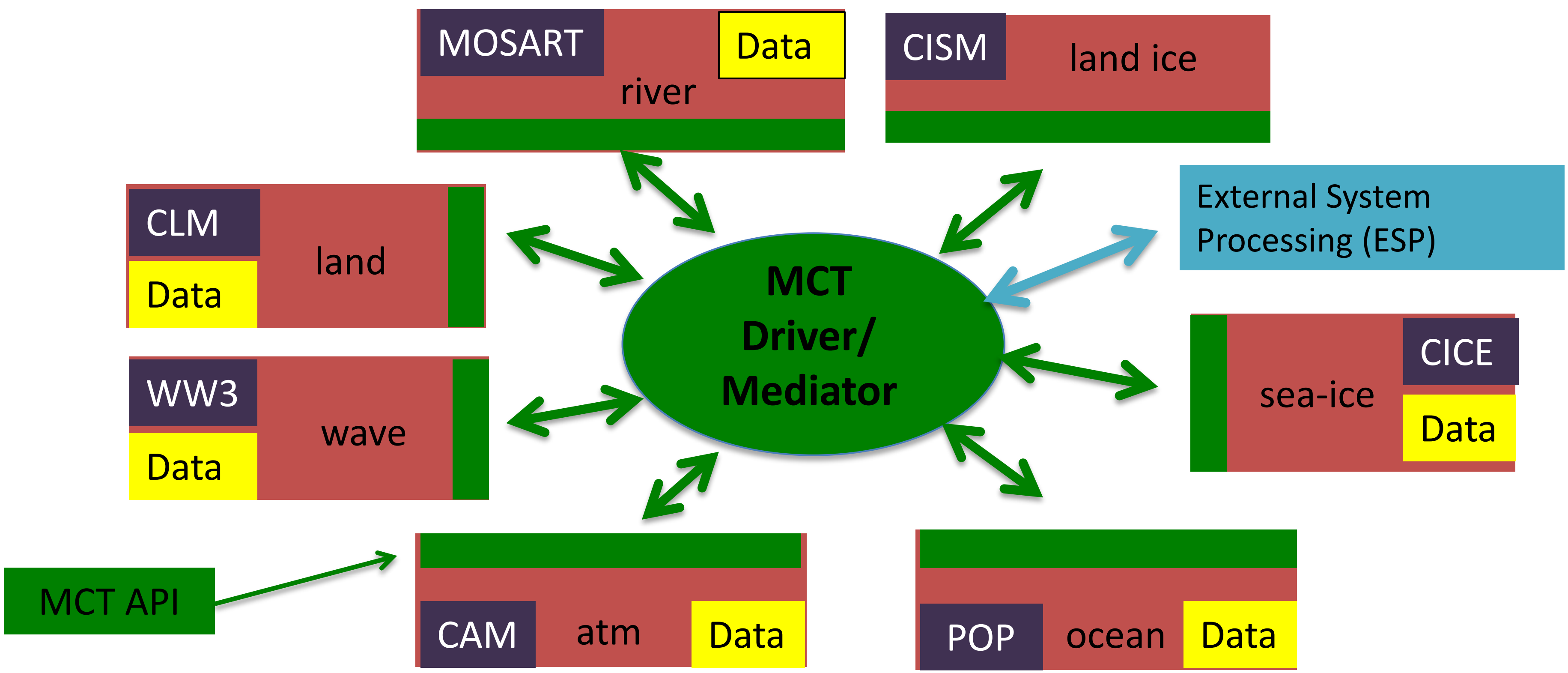
ACME/CSEG has established a very successful informal collaboration

CIME5 Update

- Major pythonization refactor effort
  - all CIME5 scripting/testing infrastructure is now in python
  - Joint ACME/CSEG collaboration
- CIME5 is no longer CESM specific
  - Share code where appropriate
  - Model specific configurability is provided
    - (e.g. $CIMEROOT/cime_config/cesm….)
  - Each prognostic component determines their own stand-alone configurations and pe-layouts
- Both ACME and CESM will adopt CIME5
  - Target is July

Current CIME5 Coupling Infrastructure

Modularity of coupling infrastructure permits users to easily switch prognostic / data components

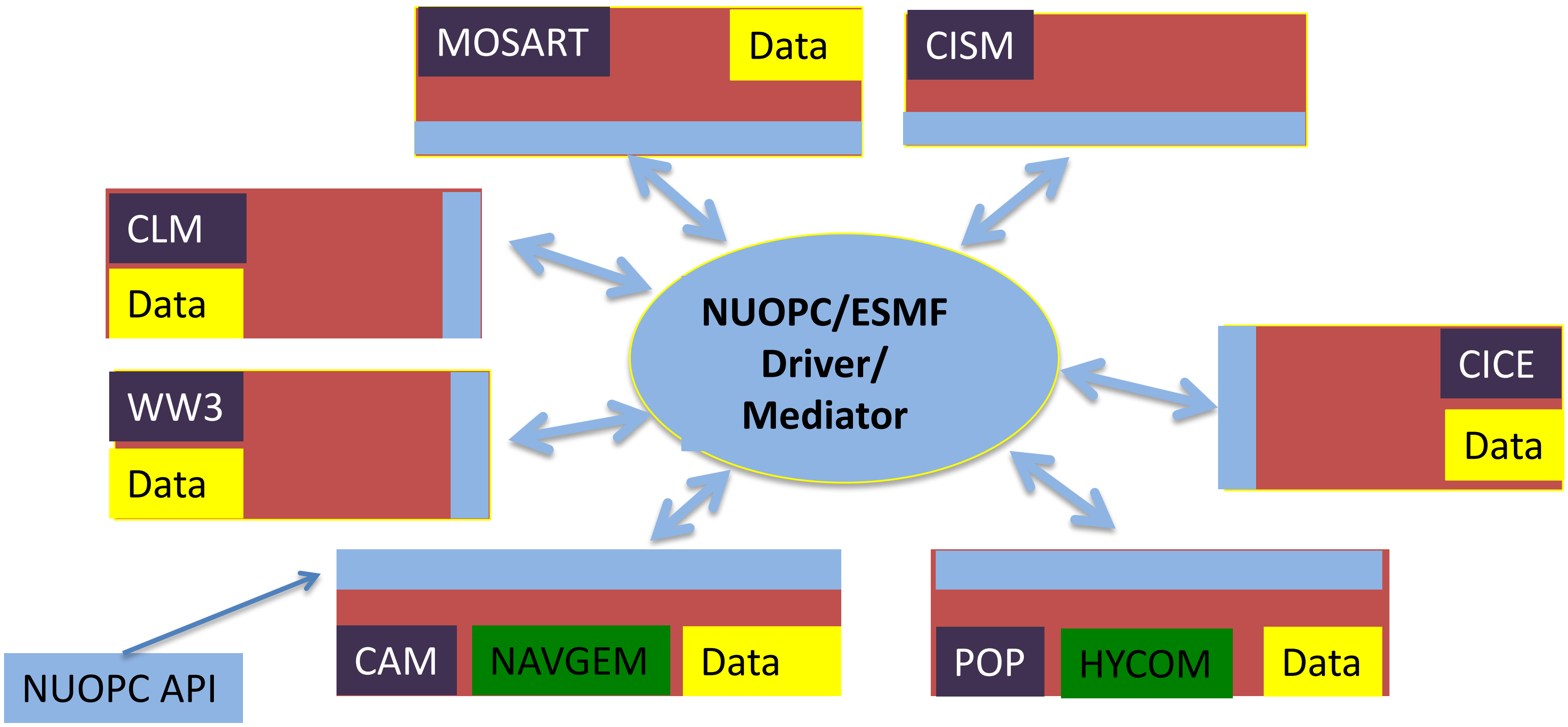


Inter-component feedbacks are easily activated and deactivated – critical for model development of target component

Note that all boxes are in CIME

Future Coupling to ESPC components

The inter-agency National Earth System Prediction Capability project (ESPC) would like to couple to CESM components



Introduction of alternative NUOPC driver-mediator in CIME will permit coupling to new prognostic components

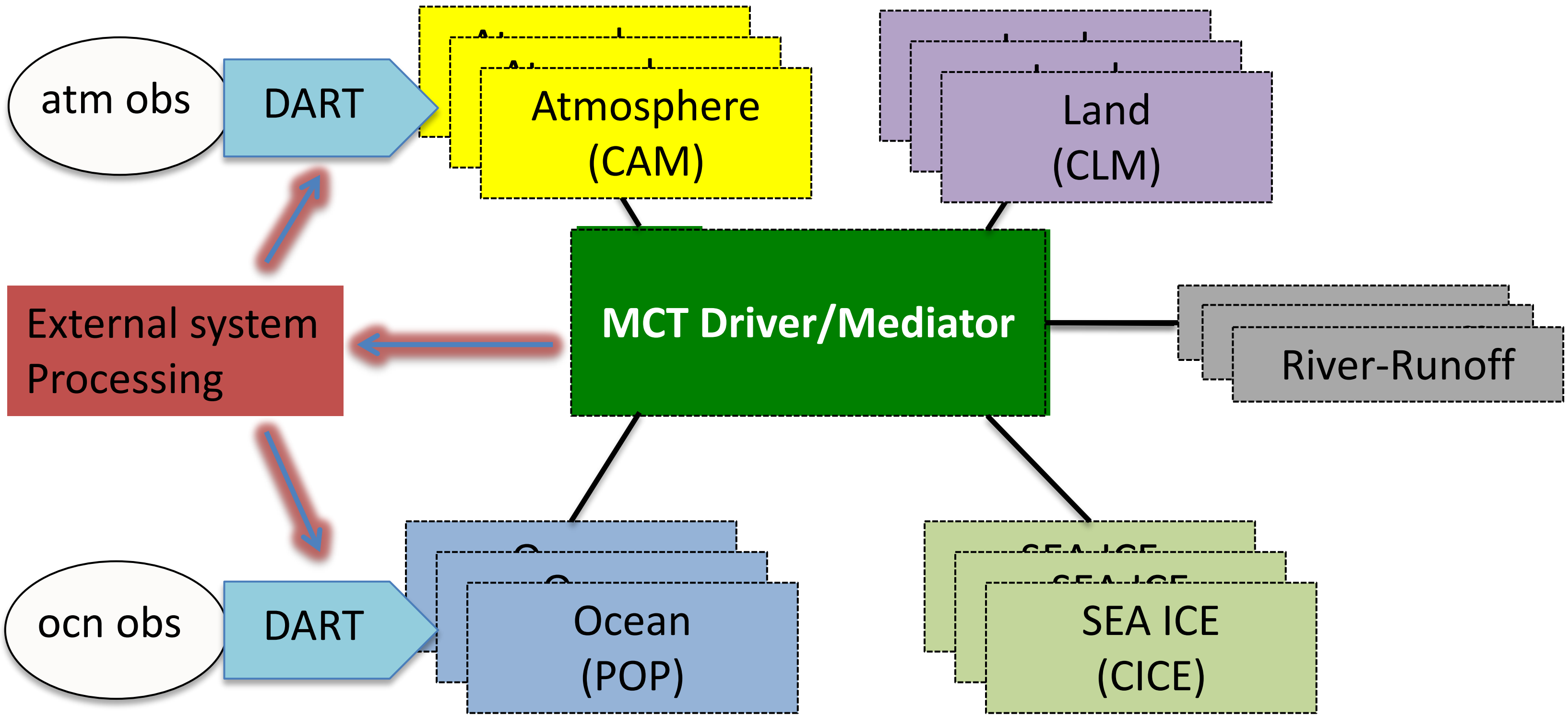
Data Assimilation in CESM

New External System Processing Component

Upcoming Pause/Resume Capability

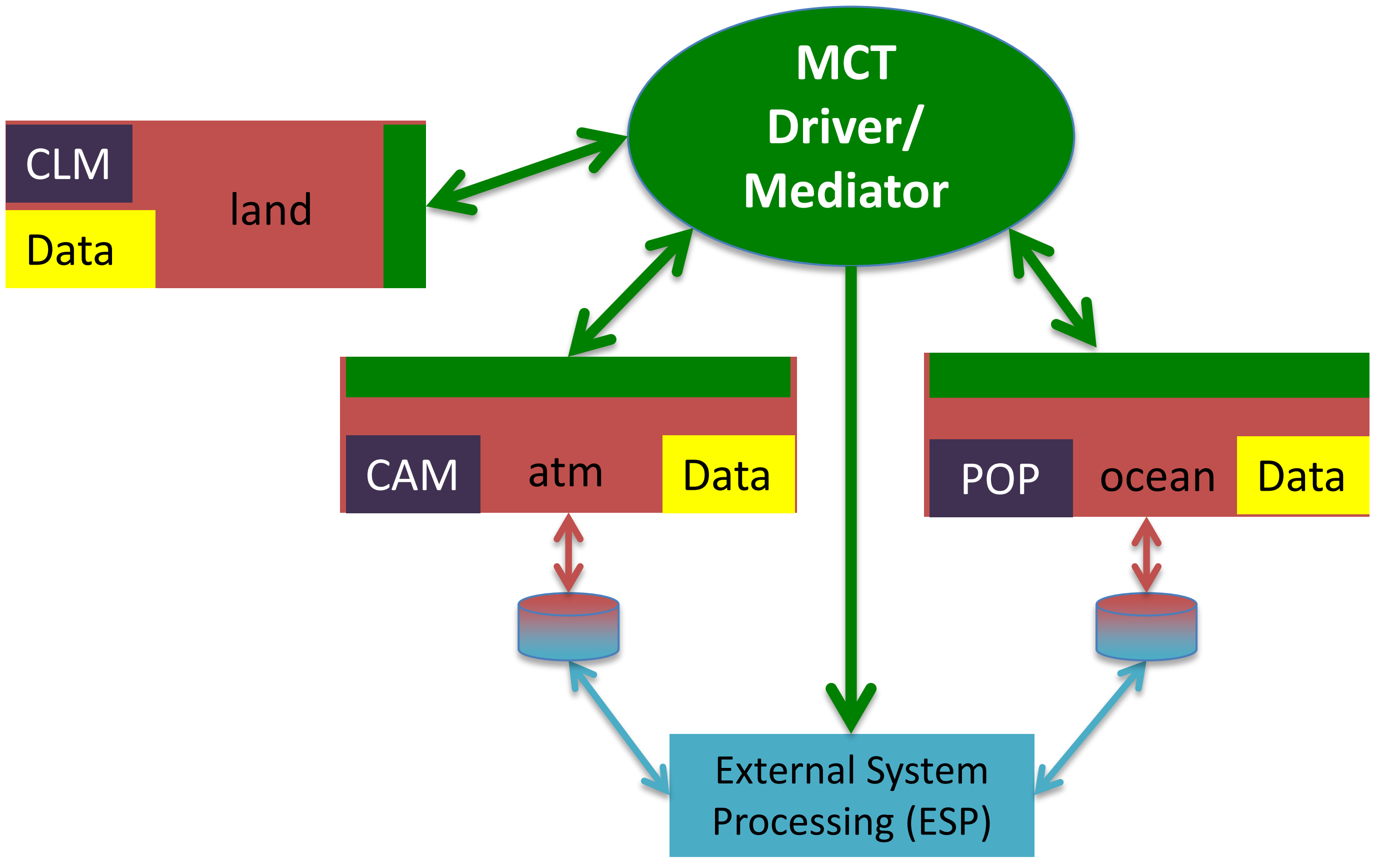
CESM Data Assimilation - New ESP capability

CIME coupling infrastructure
has unique multiple instance capability for
Kalman-Filter DART data assimilation



More Details on how ESP works

ESP component will allow data assimilation — DART — to interact directly with component data via I/O files



External tools can easily interact with CESM during simulation

New Efforts in Concurrency

Why prioritize modularity?

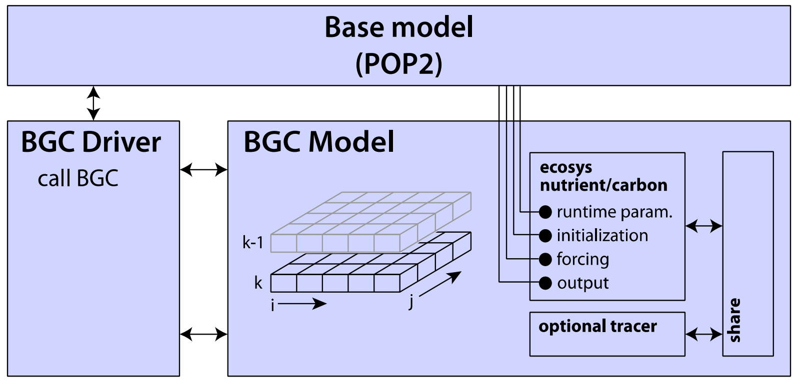
Modularity: the separation of each aspect of model functionality with self-contained code structures

- New science can be used in multiple models – removes duplication of effort
- Should not need to know about *all* the model to modify a *small* part of it
- Supports multiple alternatives for a parameterization
- Supports a hierarchy of model complexity - use certain processes, exclude others
- Supports testing individual pieces in isolation (unit testing)
- Enables addition of new concurrency

Intra-component modularization (1) : MARBL

MARBL: Marine Biogeochemistry Library

$$\frac{\partial \chi}{\partial t} + \nabla \cdot (\mathbf{u}\chi) - \nabla \cdot (K \cdot \nabla \chi) = \underbrace{B_\chi(\mathbf{x})}_{\text{MARBL}}$$



Old Model

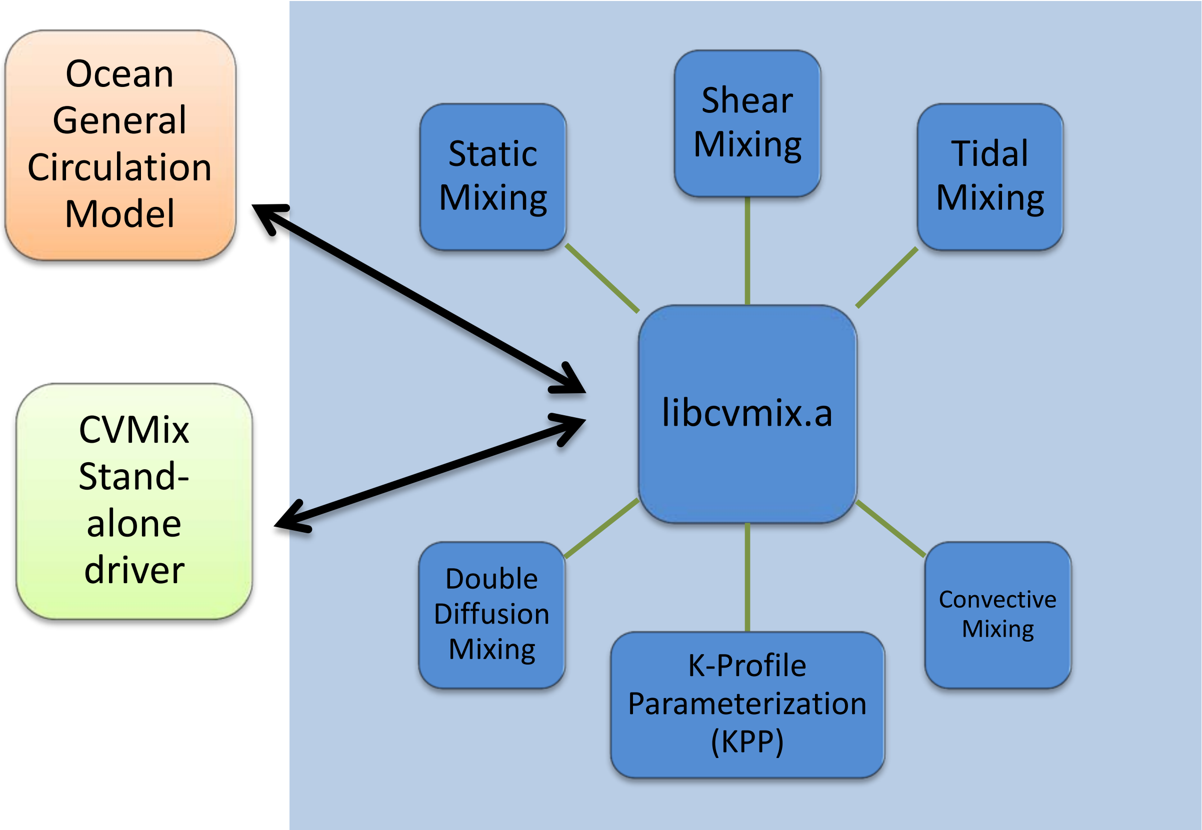
- DOE funded effort
- Enables portability to alternative physical frameworks
  - Implemented in POP – part of CESM2
  - Being implemented in MPAS-O
- Enables question dependent configurations

New Marbl-ized verison

CVMix: Community Vertical Mixing Library

CVMix used in

- POP (NCAR)
- MPAS-O (LANL)
- MOM6 (GFDL)

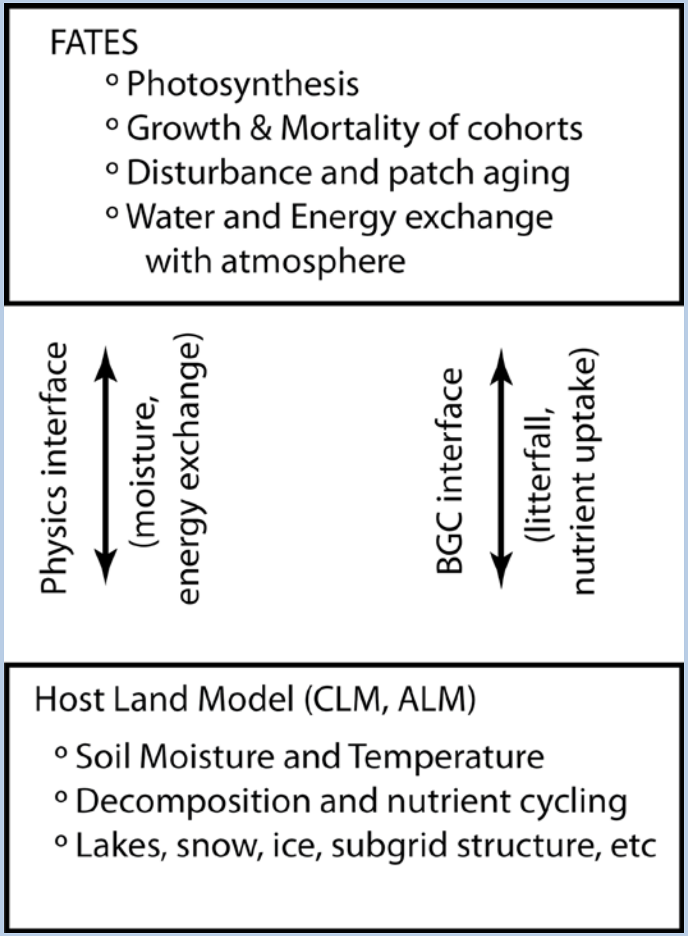


- Robust and flexible library for parameterizing ocean vertical mixing processes
- Developed within a community of scientists who make use of CVMix modules for a variety of research needs
- Ability to be used stand-alone or as part of an ocean model
- Reduces duplicate code – e.g. static mixing occurs as a step in many parameterizations

Intra-component modularization (3) : FATES

FATES: Functionally-Assembled Terrestrial Ecosystem Simulator

- Core NGEE Tropics (DOE) demographic vegetation model
- Modular interface that can be run within multiple host land models (CLM, ALM)
- Built on the CLM(ED) representation that combines ED structured vegetation representation with CLM biophysics
- Future goals:
  - Embed fine-scale laterally-resolved hydrology model to determine roles of subsurface heterogeneity and connectivity on ecosystem structure and function
  - Plan to incorporate gross land-use transitions via CLM/ALM subgrid hierarchy as transfer between distinct land-use columns

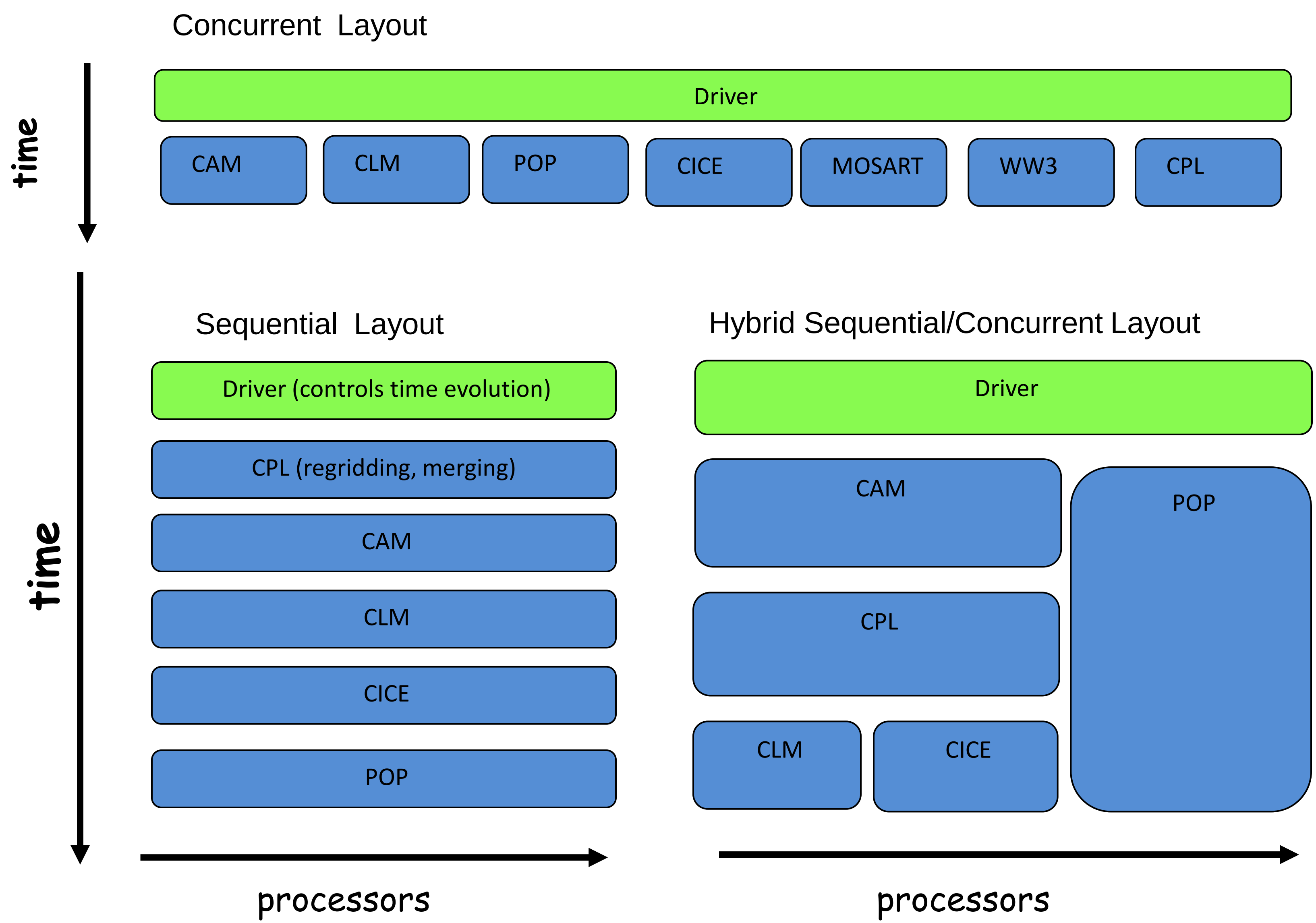


CLM Modularity Example:
Simplifying Adding New Science

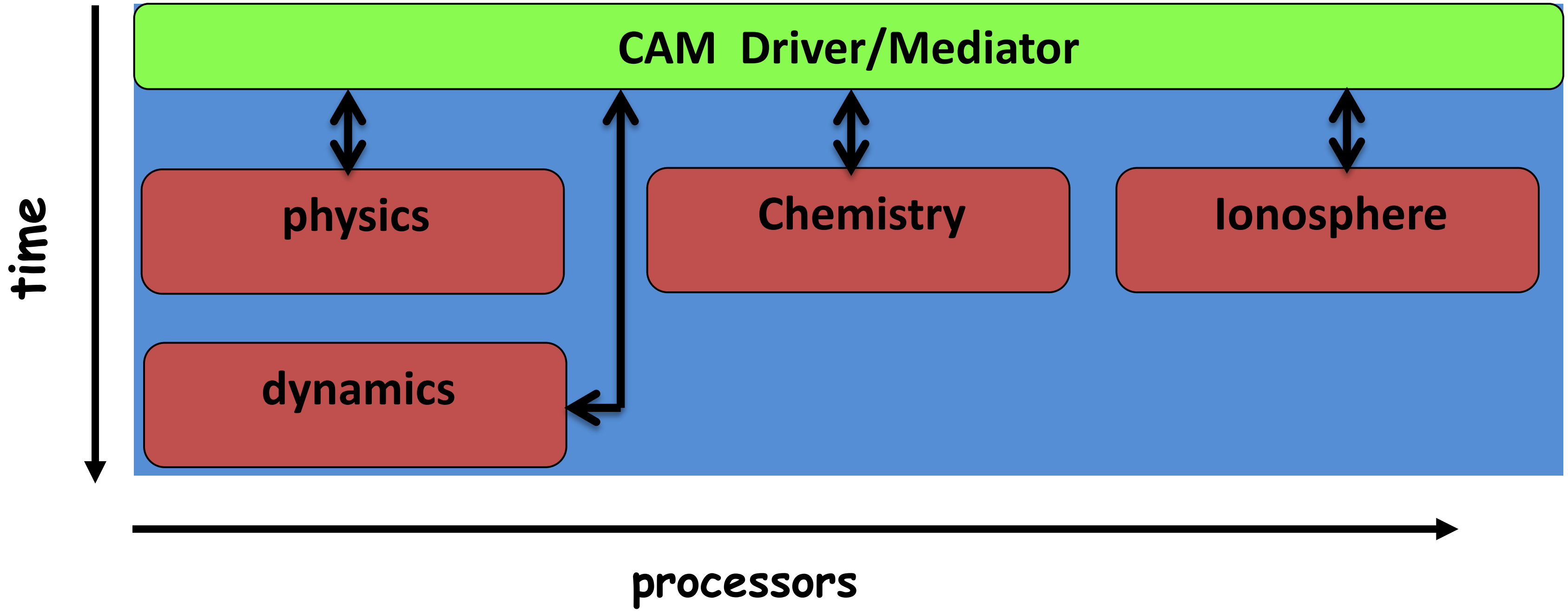
- Wood products are split 10 and 100-year lifetime pools
- Old code: **12** modules had information about these two pools
  - Adding another pool, or changing the lifetime of a pool required changes to **all 12 modules**
- New code: Only **2** modules have information about pool
  - main science module and a module that reads parameters from a file
- Result: easy to now add a pool or change the definition of the pools - all changes will be in the single science module

New Efforts in Concurrency

Inter-component concurrency



Intra-component concurrency (future)



For intra component concurrency need:

- Driver
  - control time evolution of components, calls coupling infrastructure
- Coupling infrastructure
  - carriers out redistribution and regridding
- Modularity
  - Components must be self contained modular units with APIs