

Software modernization for the UFS A position paper

<https://doi.org/10.25923/gfbx-pk53>

Hendrik Tolman
September 10, 2025



Background

- Purpose:
 - Position paper on software modernization for the UFS
 - Inventory of current activities through selected interviews
 - Encouraging formal projects on this topic
- Scope:
 - Focuses on traditional physics-based numerical model, AI/ML tools for modernization are considered relevant
- UFS Strategy:
 - Enables more efficient modernization as it limits the number of models to be considered
- This presentation:
 - Focus on outcomes / coupling

Current state and need

- UFS has matured, with many applications supported by EPIC, and with a shrinking NOAA production suite
- Environmental component model coupling is maturing, but there's a need for more integrated coupling approaches
- Existing / older code architectures are often not optimal for new hardware (exascale computing, GPUs).
- Software modernization is a long-standing issue, but has not been addressed systematically

Driving factors

- Hardware Architecture Evolution:
 - Transition from CPU to GPU.
 - Move towards exascale computing.
 - Last major refactoring was for hardware 25 years ago
- UFS issues:
 - Portability versus performance: is there a conflict?
 - Next-Generation Coupling: The current CMEPS API-level coupling (built around ESMF/NUOPC) is becoming a bottleneck for load balancing and optimization. More integrated, object-oriented coupling is needed

Key issues discussed in interviews

- Open Source and Open Science
- Fault Tolerance
- Load Balancing
- Optimization
- Coupling
- Code quality
- Knowledge and expertise
- Do we need to move away from Fortran

Outcomes 1/2

- Open-source and open-science are here to stay
- Fault tolerance is less of an issue now than 10 years ago
- There is no universally preferred method for software modernization
- AI/ML tools are crucial
- Code Quality:
 - CI/CD and DevOps positively impact code quality
 - Unit testing “+” is crucial for efficient development workflows
- Coupling of Component Models:
 - Present NEMS - ESMF/NUOPC style impedes load balancing and optimization

Outcomes 2/2

- Optimization:
 - Data structure design is critical but not universal
 - Moving away from Fortran becomes more urgent
 - Significant experience with Python and C++ and middleware / DSLs. Look at Julia or Rust?
 - Deploying cutting-edge tools on FISMA-high operational computers presents unique challenges
 - collaborate early with NCO
 - Big potential speed up with end-to-end modernization:
 - 2-5x for CPUs and 10-30x for GPUs demonstrated

Action items 1/2

Track Modernization Efforts:

- The UFS community should not mandate a detailed strategy but track existing modernization efforts as part of the annual UFS inventory.

“Owners” of UFS components are strongly encouraged to:

- Implement “decoupled” (modular) software with unit / integration / functional / regression testing in their open-science workflows.
- Strategically accelerate the transition away from Fortran
- Coordinate with operations early

Action items 2/2

Next-Generation Coupling :

- The UFS community (SAICCT ?) needs to develop a plan for the next-generation coupling approach, including:
 - Selecting a more closely coupled, community-based open-source methodology
 - Assessing if this will replace or supplement / expand the current CMEPS - ESMF/NUOPC approach
 - Evaluating feasibility of formal collaborations (e.g., NWS-OAR-NCAR MOA)

Some initial considerations

- Linkage to load balancing and general code optimization
 - Moving to localized data structures in general
 - Core parallel approach is domain decomposition for exascale computing.
- We need to move away from a CMEPS style global data exchange for exascale coupling
 - This does not mean we need to move away from ESMF
- Close coupling with different coupling time steps for different parts of a component model is needed
 - See MOM6 – sea ice coupling
- Consider National Water Model NextGen strategy
 - BMI + domain approach allows for different hydro packages used for different regions.
 - Domains with halos versus domains with boundary data
 - Also applicable to e.g., coastal versus deep ocean wave model coupling

Discussion Time

!