# Integration of MPAS Dycore into UFS

Jun Wang (NOAA/NWS/EMC), Kevin Viner (NOAA/NWS/EMC), Dustin Swales (NOAA/OAR/GSL), Dan Rosen (NCAR/CGD), Dom Heinzeller (UCAR/JCSDA), and Ligia Bernardet (NOAA/OAR/GSL)

Feedback? Send it to dom.heinzeller@ucar.edu and ligia.bernardet@noaa.gov.

A Tiger Team under the auspices of the Unified Forecast System (UFS) System Architecture and Infrastructure Cross Cutting Team (SAICCT). This work also satisfies a *Modeling Infrastructure* deliverable for the UFS-R2O Project

December 2023

Table of Contents

# Acknowledgements

The Tiger Team thanks Arun Chawla (then at EMC, now at Tomorrow.io) for his early work on this effort. It also thanks NOAA/NWS/NCEP/EMC, NOAA/GSL, DTC, NCAR/CGD, and UCAR/JCSDA for supporting this work.

# Introduction

The Unified Forecast System (UFS) Short-Range Weather (SRW) Application (App) team, along with the Rapid Refresh Forecast System (RRFS) and Warn-on-Forecasting System (WoFS) teams, have identified shortcomings of UFS-based convective-allowing predictions (Alexander et al. 2023, Wicker 2023). These problems pose a risk to the feasibility of using the current UFS code base for research and development of convective-allowing models (CAM). Extensive research was conducted to identify the causes of the problem and indications are that it relates to deficiencies in the Finite-Volume Cubed-Sphere (FV3) dynamical core (dycore) at those scales (Carley et al. 2023[1], Skamarock 2008, Konor and Randall 2018a,b). While it should be noted that FV3 has been successfully used at convective-allowing and finer scales in some research settings (Zhang 2019, Jeevanjee 2017, Jeevanjee and Zhou 2022), the UFS CAM groups have not been successful in a number of attempts to address the issues for the requirements of the UFS community. Therefore, the UFS leadership tasked the UFS Software Architecture and Infrastructure Cross Cutting Team (SAICCT), and the UFS Research to Operations Project (UFS-R2O) tasked its Modeling Infrastructure team, to scope out the technical work needed to add a second dycore to the UFS. To that end, a Tiger Team was formed to conduct this analysis and this report summarizes the team's findings.

Given that UFS SRW App, RRFS, and WoFS teams chose to explore the Model for Prediction Across Scales (MPAS) as an alternative (Alexander et al. 2023, Carley et al. 2023), the Tiger Team focused on a two-pronged approach: to scope out in broad terms the inclusion of a generic new dycore in the UFS and to focus the majority of its work on the MPAS dycore. Similarly, since the drive for a new dycore comes from the UFS SRW App, RRFS, and WoFS teams, the Tiger Team kept in mind the use of the MPAS dycore for all UFS Apps, while focusing primarily on the UFS SRW App, RRFS, and WoFS. The configurations of the convective-allowing modeling systems today are limited-area models with an atmospheric-only component (with an embedded columnar land surface model and smoke and dust representation), optionally coupled with the atmospheric composition component Community Multiscale Air Quality (CMAQ) model.

The Tiger Team was in effect from late August to December, 2023. Its members, the authors of this document, were not supported explicitly for this endeavor - they volunteered their time to meet for 30 min weekly during this period. The depth of details in this report is commensurate with the amount of time we were able to dedicate. In addition to internal group meetings, the Tiger Team collected input from the following external sources: leadership of the UFS MRW/S2S, SRW, and Hurricane Apps, MPAS representatives Bill Skamarock and Michael Duda (National Center for Atmospheric Research, NCAR, Microscale and Mesoscale Meteorology Laboratory; MMM), and the co-chair of the Atmospheric Model Working Group within the

---

[1] While this white paper has only been distributed to a selected group, our understanding is that it will soon be publicly available.

Community Earth System Modeling System (CESM), Peter Lauritzen (NCAR Climate and Global Dynamics Laboratory; CGD).

The connection with both NCAR MMM and CGD is relevant because MMM develops and uses the MPAS model, while CGD uses the MPAS atmospheric component (MPAS-A) dycore in the Community Atmospheric Model, the atmospheric component of CESM. The solution sought by the Tiger Team is similar to the one used by CGD, that is, the vision is to use the MPAS-A dycore without using the entire MPAS-A. This arrangement will allow the UFS to retain core parts of its infrastructure, such as its workflow, the connection with physics via the Common Community Physics Package (CCPP), I/O, post processing capability, and product generation.

This report is organized in multiple sections, corresponding to the various areas of work needed to bring a new dycore into the UFS. The report concludes with an estimation of resources for adding the MPAS dycore to the UFS and a section wrapping up the message and indicating the way forward.

# Generalizing the Atmospheric Component

The atmospheric component of the UFS Weather Model (WM), FV3ATM, acts as a coupling interface for interactions among dynamics, physics, atmospheric Input/Output (I/O), and external components (such as the ocean, sea ice and wave models, etc). Some portions of code tie directly to FV3 dynamics, and these portions should be abstracted to support multiple dycores (FV3 and MPAS, in particular). Additionally, FV3ATM has technical debt that has accumulated over the years through the organic addition of features. This growth leads to maintenance issues and lower extensibility.

The generalization and refactoring of the atmospheric component will facilitate the introduction of a new dycore (such as MPAS) and enhance performance. One specific upgrade that has a high probability to boost performance is a change from blocked data structures to contiguous arrays. This is because unnecessary copies of data in the physics *timestep_init* phase can be avoided and transforming data to what the dycore expects will be more efficient.

The reorganization of the atmospheric component, renaming from FV3ATM to UFSATM, is a prerequisite for the connection between MPAS, the CCPP, and external components. Figure 1 below shows the code structure change. Left is the current atmosphere component (FV3ATM) and the right panel is for the proposed atmosphere component UFSATM.

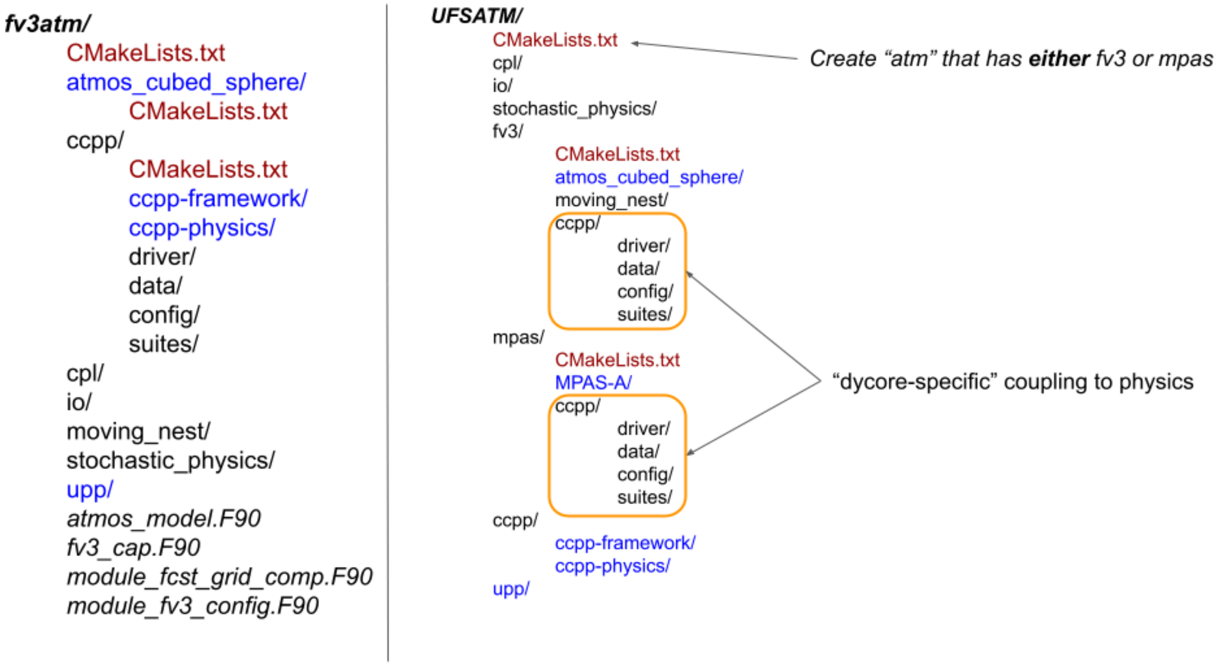

*Figure 1. Organization of codebase for the atmospheric component of the UFS WM, showing the a) current organization (FV3ATM) (left) and the proposed organization (UFSATM) with MPAS (right). Subcomponents are highlighted in blue and Cmake lists in red.*

# Code Management and Testing

The MPAS-A dycore is not isolated to its own repository. Instead, it is embedded within the general MPAS repository (https://github.com/MPAS-Dev/MPAS-Model). Therefore, the entire MPAS code will be brought into the UFS, but only the dycore and a few utilities will be used.

To integrate MPAS into the UFS WM, the following software needs code management based on the new structure of the atmosphere component discussed in the previous section:

- **UFS-MPAS interface**: will be located in the atmospheric component (UFSATM) repository of the UFS WM and managed by the UFS community.
- **MPAS:** will be a submodule under the UFSATM pointing directly to the MPAS official repository. It will continue to be managed by the MPAS team at NCAR. It is expected that the MPAS integration into UFS may require some changes and additions to MPAS such as additional coupling fields, IO etc..
- **MPAS utilities**: tools such as MPAS *pool routines* and *convert_mpas,* will be submodules to ufs_util repositories and be managed by the MPAS team at NCAR MMM. It is expected that the MPAS integration into UFS will require some changes and additions to *convert_mpas* (see the post processing section).

Testing is critical to ensure a smooth and correct integration of MPAS into the UFS and to maintain the MPAS-related features in the future UFS WM development. The following testing tasks are needed:

- Create unit tests for new functions or features.
- Create atmosphere-only regression tests with incremental updates.
- Update code quality test protocol to run MPAS in debug mode and ensure reproducibility.
- Conduct the tests above when pull requests are submitted.

# Build System

The UFS WM relies on the CMake software to manage its build system. Each component of this coupled system, and any subcomponents (e.g. dynamical core and atmospheric physics), have their own Cmake steps to build their respective code. As the UFS WM is generalized from FV3ATM to UFSATM, the level at which the Cmake lists are placed changes.

## Dependencies within MPAS-A

The MPAS-A requires the Parallel I/O (PIO) library, which itself requires standard and parallel netCDF libraries. These libraries are already included as part of spack-stack, the software stack used for UFS applications. Therefore, the addition of the MPAS dycore to the UFS does not require stack modifications.

# Pre-processing

MPAS uses an unstructured grid, or mesh, for its spatial discretization. The files describing the MPAS mesh need to be generated offline and a priori to running the model. Currently the tools to generate these scripts are not publicly available for users to create their own meshes. Instead, MPAS meshes are created on request by NCAR/MMM. NCAR/MMM has plans to make the grid generation tools available for public use. Scripts to create regional meshes from a larger global mesh are provided with MPAS.

Once the MPAS grid-describing files are available, already-existing MPAS utilities can be used to prepare initial-condition and static files for use with MPAS.

# Data Assimilation

The following is based on the assumption that no efforts will be made to interface the currently operational GSI with the MPAS dycore. Instead, all efforts will focus on the integration of MPAS with JEDI within the UFS. For JEDI, the bulk of the work on the data assimilation (DA) side is already taken care of: The JEDI DA algorithms and observation operators are developed in a model-agnostic way that allows for easy swapping of the model interface. Both the FV3 and MPAS dycores have been interfaced to JEDI by JCSDA and its partner agencies. Different DA algorithms, including 3DVar, 3DEnVar, 4DEnVar, and LETKF, have been demonstrated with both interfaces to JEDI.

JEDI uses grids provided by model interfaces for DA, i.e. it can do DA on the native grid for both FV3 and MPAS. Experiments are conducted with JEDI-FV3 (FV3 dycore only), JEDI-UFS and JEDI-MPAS on a regular basis by JCSDA and the respective partner agencies (NOAA-NWS/OAR, NASA-GMAO, NCAR-MMM). The interface between JEDI and the atmospheric component of the UFS is implemented via NUOPC, i.e. JEDI acts as a NUOPC component that communicates with the atmospheric component of the UFS, similar to how CMAQ communicates with the atmospheric model. The interface code currently resides in separate branches of the UFS WM repository and its submodules, and is maintained by JCSDA. An effort is underway to merge these interfaces into the authoritative UFS WM repository and to set up a test within the UFS WM regression testing system to ensure it remains functional.

# Physics-Dynamics Coupling

The physics-dynamics coupling happens inside the atmospheric component (now FV3ATM, to become UFSATM), with the exception of the stochastic physics (Fig. 2). The first part of the UFS physics suite, comprised of the parameterizations for radiation, surface layer, surface (land, ocean, and sea ice), boundary layer, and Rayleigh damping, is computed using a hybrid of parallel and sequential splitting described in Donahue and Caldwell (2018), a method in which the various parameterizations use the same model state as input but are impacted by the preceding parameterizations. The tendencies from the various parameterizations are then added together and used to update the model state. The second part of the UFS physics suite, comprised of the parameterizations of gravity wave physics, ozone, stratospheric water vapor, deep and shallow convection (if using), and microphysics, is computed using sequential splitting in the order listed above, in which the model state is updated between calls to the parameterization. If the GFDL microphysics parameterization is used, saturation adjustment is invoked at shorter timesteps along with the dynamical solver. In MPAS (Fig. 3), a timestep starts with all parameterizations, except microphysics, ingesting the same model state and producing tendencies. Then the state is updated during a third-order Runge-Kutta (RK3) step using the tendencies computed in the current timestep plus the microphysics heating tendencies from the previous timestep. Next, saturation adjustment is computed and the model state is updated again. Tendencies from coupling, if used, also need to be taken into account. Since not all parameterizations used by RRFS return tendencies, it will be necessary to update selected parameterizations in CCPP Physics to do that.

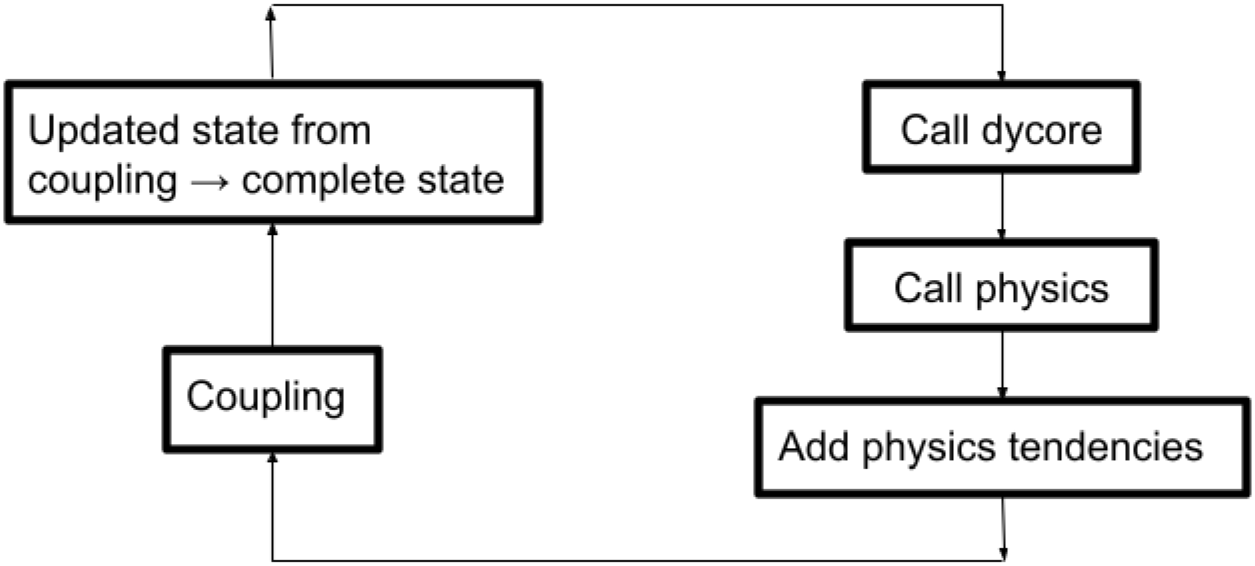

*Figure 2. Simplified FV3 physics-dynamics coupling.*

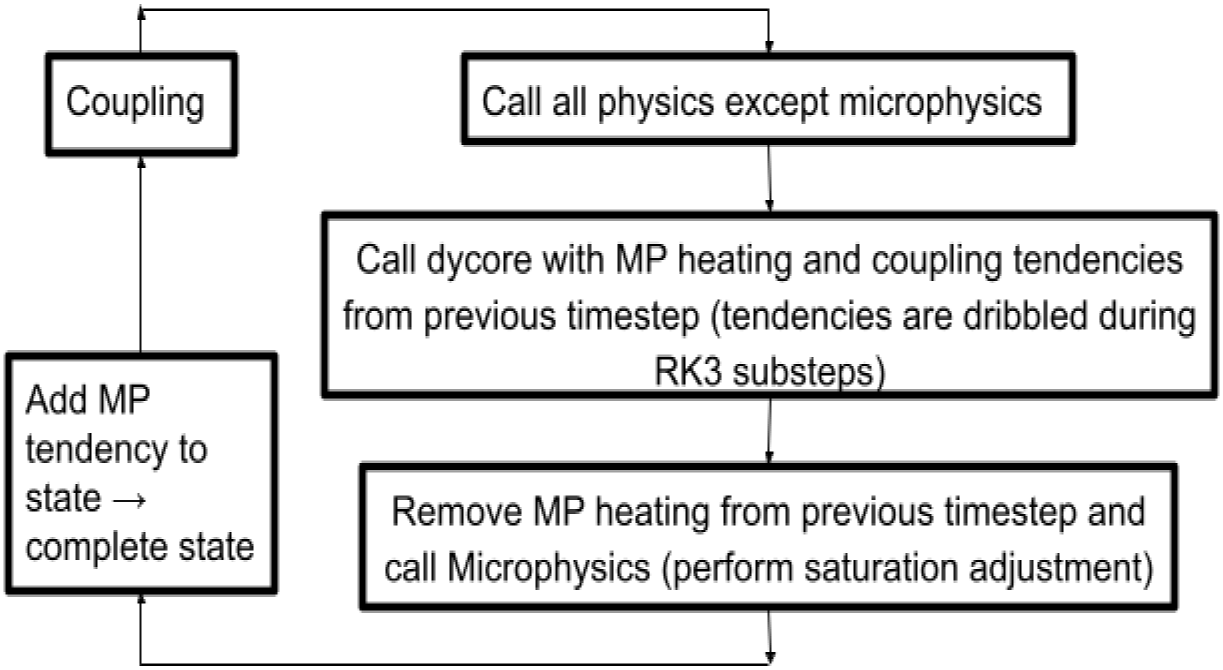

*Figure 3. Simplified MPAS physics-dynamics coupling.*

The following tasks need to be completed in UFSATM and its subcomponents, and in stochastic physics

- Update parameterizations in CCPP Physics to return tendencies or update the model state in accordance with the MPAS physics-dynamics coupling strategy.

- Update calling logic in UFSATM: different group of physics at different times (before/after dycore); in particular passing previous-timestep microphysics tendencies to dycore, together with current-timestep tendencies of other physics, letting the MPAS dycore update the state, and then performing the saturation adjustment in the microphysics after the call to the dycore.
- Perform variable transformations to account for different thermodynamic assumptions made between the MPAS dycore (dry and at constant volume) and the physics (moist and at constant pressure).
- Perform coordinate transformations between pressure levels (CCPP Physics schemes) and geometric height (MPAS).
- Perform array shape transformations from (i,k) to (k,i).
- If needed, perform unit conversions for variables for which MPAS and the physics use different units.
- UFSATM should use the MPAS *pool routines* to access the dycore fields.
- Update or replace FV3-specific interstitials in CCPP Physics so they are either generalized or specific for MPAS.
- Update CCPP suites with new interstitials.

Note that some of the conversions listed above are already available in MPAS and need to be called by UFSATM (similar to how this is done for the FV3 dycore)

Regarding stochastic physics, it is expected that SKEB, SHUM, SPP, and SPPT will require minor changes because they are somewhat hard coded to the *GFS data types*. Since the spatial pattern for those tools is not tied to the FV3 native grid, it can be connected to MPAS without major work. The current cellular automata code won't work with MPAS' irregular mesh unless additional work is done. However, this may not be needed for RRFS since it is not used in RRFS v1 and its impact is more important for longer time scales.

# Inter-Component Coupling

Current inter-component coupling in the UFS is handled through NUOPC interfaces. Each component, UFSATM (FV3+CCPP+I/O), AQM, CDEPS, CICE, CMEPS, GO KART, HYCOM, MOM6, NOAHMP, and WW3 provides a standardized coupling interface, which is commonly referred to as a NUOPC cap. Each cap is maintained independently and the changes needed to couple atmosphere with MPAS dycore will be made exclusively within UFSATM. Development of the inter-component coupling capabilities should be done after the generalization of the atmospheric component.

The work involved includes:

- Exposing the MPAS dycore geometry to the UFSATM cap, including: global and regional grids and meshes.
- Exposing the MPAS dycore domain decomposition to the UFSATM cap. This allows the dycore to parallelize itself in a decomposition best fitted for the model

- Driving the MPAS dycore through the UFSATM cap. This includes calling data memory allocation and value initialization in a multiple phase initialization sequence, calling one or more run phase routines, and finalizing the model.
- Connecting import and export fields from UFS ATM cap to the MPAS dycore. There are currently 71 export fields from UFSATM and 16 import fields to UFSATM. These fields are a combination of layered atmosphere fields, surface fields, layered soil fields, and tracers. Some of these fields pass through/to the dynamical core and some pass through/to the physics.
- Verifying that the atmospheric forecast portion of UFSATM continues to work with the IO portion of UFSATM.

Work to complete the inter-component connection can leverage expertise from the coupling between the  CESM Community Atmospheric Model and the MPAS dycore, which also utilizes NUOPC.

# Input/Output

Currently MPAS outputs three types of files, all on the native mesh: diagnostic files, history files and restart files. The diagnostic files contain pressure level fields computed by the model. The history and restart files contain model state fields on native unstructured centroidal Voronoi mesh in C-grid and a geometric-height vertical coordinate. MPAS has the option to use PIO with namelist variables to control the IO performance. The *convert_mpas* utility is available to interpolate these output files from the native mesh to lat-lon grids. A fully asynchronous I/O can be implemented in two steps, while the first step will allow applications to produce history and restart files for downstream jobs.

**Step 1: Use the MPAS I/O capability.**
In this method, MPAS is responsible for generating required diagnostic fields. Since UFS applications, such as RRFS, create a wide range of products, it is recommended that only a minimal set of products be generated at this step. This avoids adding tremendous UPP code into MPAS to generate a full set of POST products. Two subtasks are required:

- Output history and restart files on the MPAS native mesh directly.
- Use *convert_mpas* to convert the history and restart to lat-lon grids. This task may require changes in *convert_mpas* to add additional regional grids desired for UFS products.

**Step 2: Develop asynchronous I/O (write grid component) capability in MPAS.**
This work requires adding code updates to allow forecast tasks to prepare the output data in ESMF field bundles that write grid components can pick up. The code updates can be implemented without impacting the current IO capability in MPAS. The following subtasks are required to complete Step 2:

- Set up output field bundles in the MPAS dycore driver code.
- Set up the write grid component in the atmosphere cap.
- Implement redistribution method required to output history files on native mesh.
- Implement interpolation method required to output history files on lat-lon or other grids.

# Post-Processing

As previously mentioned, MPAS outputs diagnostic, history, and restart files on its native mesh. Those files can be interpolated using the *convert_mpas* utility from the native mesh to a lat-lon grid. This is the simplest way to get the post processing fields from the diagnostic files. However, to enable a full list of post processed products for UFS applications, an incremental approach is needed. The steps are:

**Step 1: A minimal set of products generated using MPAS I/O and tools.**
In this step a minimal set of products that developers can use for an initial evaluation of the MPAS dycore in the UFS is produced. MPAS directly outputs diagnostic files with post-processed products. Then *convert_mpas* is used to convert diagnostic files from native to lat-lon grid.

Necessary development:

- Integrate the MPAS dycore (and CCPP physics) in UFS atmosphere grid component (see previous sections)

**Step 2: A comprehensive set of products is generated by using MPAS and UPP.**
In this step the list of products is augmented from simple diagnostics to a full set of products by incorporating the offline Unified Post-Processor (UPP). MPAS directly outputs full model states in the history files on its native grid. The *convert_mpas* utility will convert the history files to lat lon grid. The offline UPP will then read the history files and create a comprehensive set of products on lat lon grid.

Necessary development:

- Update UPP to create postprocessed products in isobaric or other vertical levels from model states in the MPAS vertical coordinate.

**Step 3: Faster post-processing with inline UPP on the write grid component.**
In this step, postprocessing is done through the use of inline UPP on the write grid component. Products will be created faster since post-processing will run simultaneously with the UFS WM on a dedicated set of processors and will not require reading intermediate files for post processing. In this step interpolation will be done between the forecast grid component and write grid component, this allows the model to output the history files and post product on other grids besides lat lon grid.

Necessary development:

- Write grid component needs to be developed (see the previous section) to interplate the the model states from native MPAS grid to current UFS application (RRFS) product grid.

# Workflow

The primary workflow development pertains to the integration of the MPAS preprocessing tasks described above. While JEDI also needs to be added to the workflow, that work is agnostic of whether the FV3 or MPAS dycore is used.

The resources needed for workflow development are somewhat difficult to scope out since there are many different workflows in use by the UFS (approximately one per UFS application). As far as UFS CAM applications are concerned, there are at least three workflows to consider: the UFS SRW App workflow (used by the community and supported by EPIC), the RRFS workflow (used by GSL and EMC and soon to be operational), and the Unified Workflow (under development by EPIC and GSL). Additional workflows, such as EWOK used by JEDI-Skylab at JCSDA, can also be considered since it can handle DA and forecasting tasks for both dynamical cores. However, the EWOK code is not openly available and does not comply with NCO implementation standards, so programmatic and technical work would be needed to use EWOK.

While the UFS SRW App and RRFS leadership have expressed the intention to test MPAS for RRFS v2 operations, it is necessary to integrate MPAS into UFS SRW App and RRFS without losing the capability to run the FV3-based UFS WM. The MPAS-based UFS WM will need to be tested before a decision about the way forward can be made. Generalizing the workflows such that they can be run for the MPAS or FV3 dycore will demand more resources than forking a workflow, customizing it for MPAS, and maintaining the deltas.

# Level of effort estimates

The level of effort listed in Table 1 is for enabling the UFS SRW App (and its RRFS operational instantiation) to run with the MPAS dycore in atmosphere-only mode, using a configuration similar to the one used by RRFS today. It does dot include scientific testing.

Estimating levels of effort is a difficult task because of unknowns that are uncovered during the development process, which can make the work easier or harder. The resources are also highly dependent on the expertise of the engineers doing the work, as well as and on their level of familiarity with the UFS and MPAS. The estimates provided below are conservative numbers for expert staff.

The estimates in Table 1 only include effort needed by the UFS team. It is expected that the MPAS team at NCAR may need resources for this collaboration.

*Table 1. Tasks necessary for inclusion of the MPAS dycore in the UFS. Resources as expressed as Full-Time Equivalent (FTE) for a year. The Notes column contains information about dependencies among the tasks*

| Task | FTE | Notes |
|---|---|---|
| Code Mgmt and Governance at NOAA | 0.5 | Ongoing |
| Generalizing ATM: FV3ATM → UFSATM | 1.0 | |
| Build system and software stack | 0.6 | |
| Pre-processing | 0 | |
| Data Assimilation | 0 | |
| Post-processing/IO | 4.0 | Dependent on generalized UFSATM |
| NUOPC Cap[2] (inter-component coupling) | 1.0 | Dependent on generalized UFSATM |
| Connection to CCPP (physics dynamics coupling) | 2.0 | Dependent on generalized UFSATM |
| Workflow | 4.0 | Partially dependent on the other elements being ready to run |
| **Total** | **13.1** | |

# Summary

The Tiger Team estimated that about 13 FTEs are needed to update the UFS to have the capability to use the MPAS dycore for a configuration similar to RRFS v1 (regional, atmosphere only). This estimate does not include scientific testing and evaluation. The estimate has large uncertainty, mostly related to the expertise and experience of the team that would do the integration. It is recommended that the estimates be reviewed, and possibly updated, by the organization(s) resourced to do the work.

For the success of this endeavor, it is paramount to establish a collaborative framework with the MPAS development team at NCAR, so that code updates can be continuously exchanged even after the MPAS-UFS integration. The estimates presented do not include funding possibly needed by NCAR to partner with the UFS in this effort.

---

[2] https://earthsystemmodeling.org/nuopc/

Whenever possible, an effort was made to indicate which development work is foundational (namely, the generalization of the FV3ATM component) versus development that depends on previous tasks. Ongoing work, such as code management, was called out separately. The material was presented in this way to facilitate the creation of a project with multiple simultaneous efforts.

Finally, the estimates about the workflow effort should be considered as the most uncertain since it is not clear whether development will be needed on a single workflow (e.g., SRW App) or multiple workflows. Certainly the consolidation of the development onto a single workflow will minimize the cost. Additionally, the workflow estimate depends on whether a truly integrated workflow would be created, for running either the FV3 or the MPAS dycore with the flip of a switch, or whether separate workflows would be maintained. The decision may depend on the intention of retaining one or both dycores in the long-term.

# References

Alexander, C., J. Carley, and M. Pyle, 2023: The Rapid Refresh Forecast System: Looking Beyond the First Operational Version. Unifying Innovations in Forecast Capabilities Workshop, July 24-28, Boulder, CO.

Carley, J., C. Alexander, L. Wicker, C. Jablonowski, A. Clark, J. Nelson, I. Jirak, and K. Viner, 2023: A report on mitigation efforts to address Rapid Refresh Forecast System (RRFS) v1 dynamical core performance issues and recommendations for RRFS v2. Under review.

Jeevanjee, N., 2017: Vertical velocity in the gray zone. J. Advances Modeling Earth Systems, 9(6), DOI:10.1002/2017MS001059.

Jeevanjee, N., and L. Zhou, March 2022: On the resolution-dependence of anvil cloud fraction and precipitation efficiency in radiative-convective equilibrium. J. Advances Modeling Earth Systems, 14(3), doi:10.1029/2021MS002759.

Konor, C. S., and D. A. Randall, 2018a: Impacts of the horizontal and vertical grids on the numerical solutions of the dynamical equations – Part 1: Nonhydrostatic inertia–gravity modes. *Geosci. Model Dev.*, **11**, 1753-1784, https://doi.org/10.5194/gmd-11-1753-2018.

Konor, C. S., and D. A. Randall, 2018b: Impacts of the horizontal and vertical grids on the numerical solutions of the dynamical equations – Part 2: Quasi-geostrophic Rossby modes. *Geosci. Model Dev.*, **11**, 1785-1797, https://doi.org/10.5194/gmd-11-1785-2018.

Skamarock, W. C., 2008: A Linear Analysis of the NCAR CCSM Finite-Volume Dynamical Core. *Monthly Weather Review*, **136**, 2112–2119, https://doi.org/10.1175/2007MWR2217.1.

Wicker, L., 2023: Assessment of Convective-Scale Attributes of the FV3 Dycore Using Idealized Simulations. Unifying Innovations in Forecast Capabilities Workshop, July 24-28, Boulder, CO.

Zhang, C., and coauthors, 2019: How Well Does an FV3-Based Model Predict Precipitation at a Convection-Allowing Resolution? Results From CAPS Forecasts for the 2018 NOAA Hazardous Weather Test Bed With Different Physics Combinations. *Geophysical Research Letters*. **46.** https://doi.org/10.1029/2018GL081702.