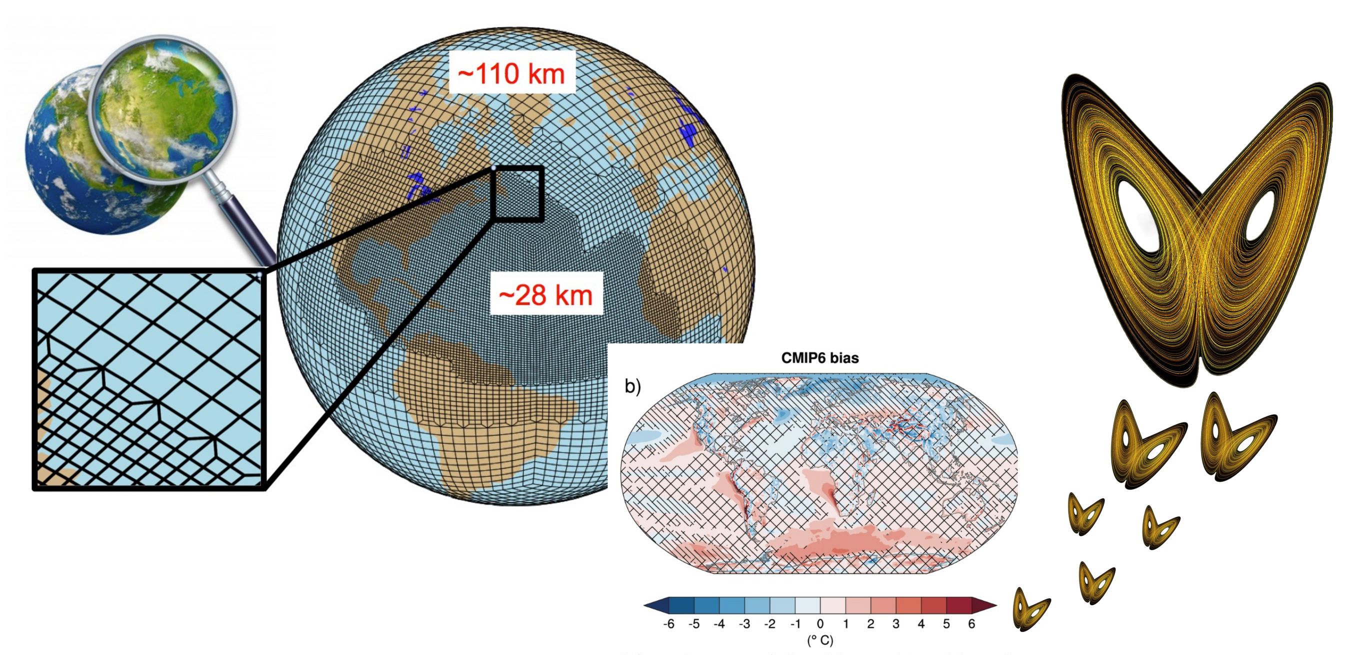
DART-NUOPC: A first prototype for in-memory data transfer between Earth System Models and Data Assimilation

Suman Shekhar[1], Anh Pham[2], Helen Kershaw[3], Dan Amrhein[3], Ufuk Turuncoglu[3]

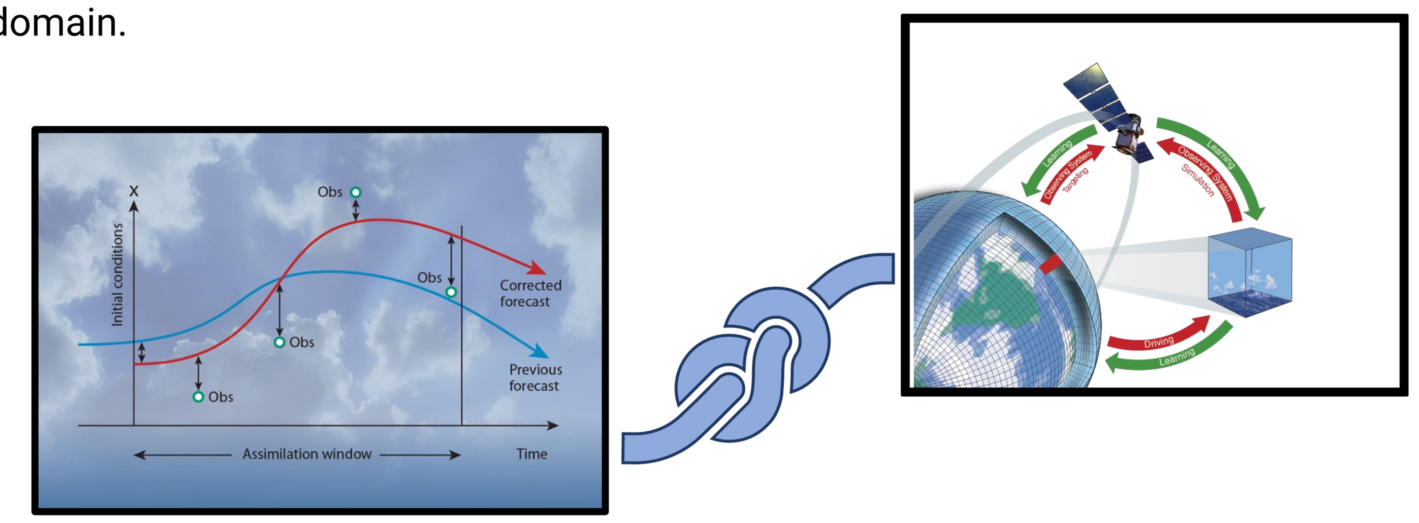
[1]*Rutgers University, The State University of New Jersey,* [2]*Mount Holyoke College, Massachusetts ,* [3]*NSF National Center for Atmospheric Research (NCAR)*

contact – suman.shekhar@marine.rutgers.edu

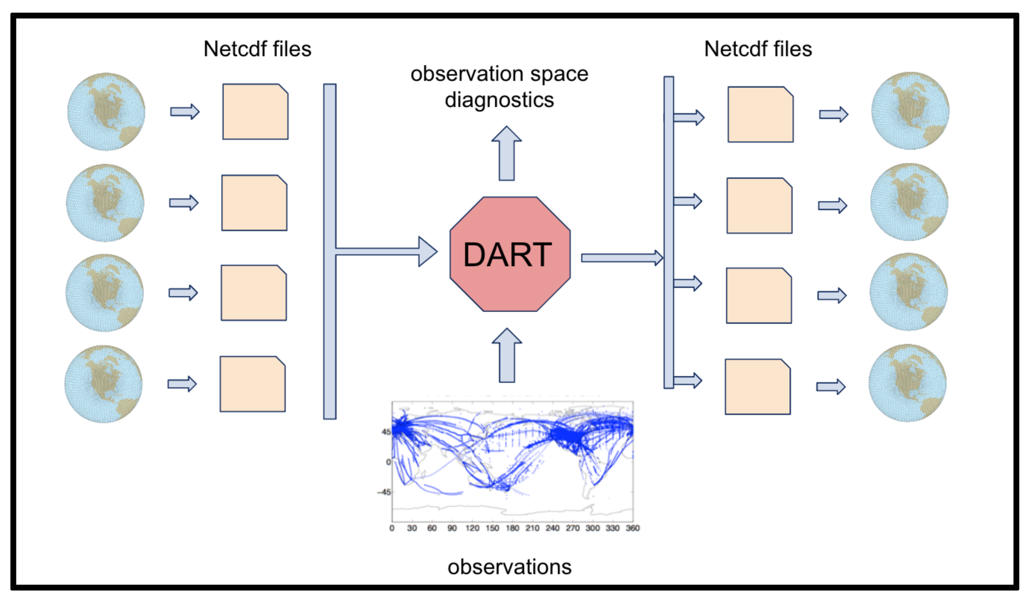
BACKGROUND



Climate change is a complex problem and making future predictions is challenging, not just because of its indeterministic nature hiding behind non-linear partial differential equations, but also because the models often exhibit biases due to missing physics and coarse model grid representations of the domain.



We need both observational data and algorithms to integrate this data into the model predictions. Data assimilation (DA) is an algorithm which integrates observational data with model predictions and are critical for mitigating these biases and steering models towards more accurate states.



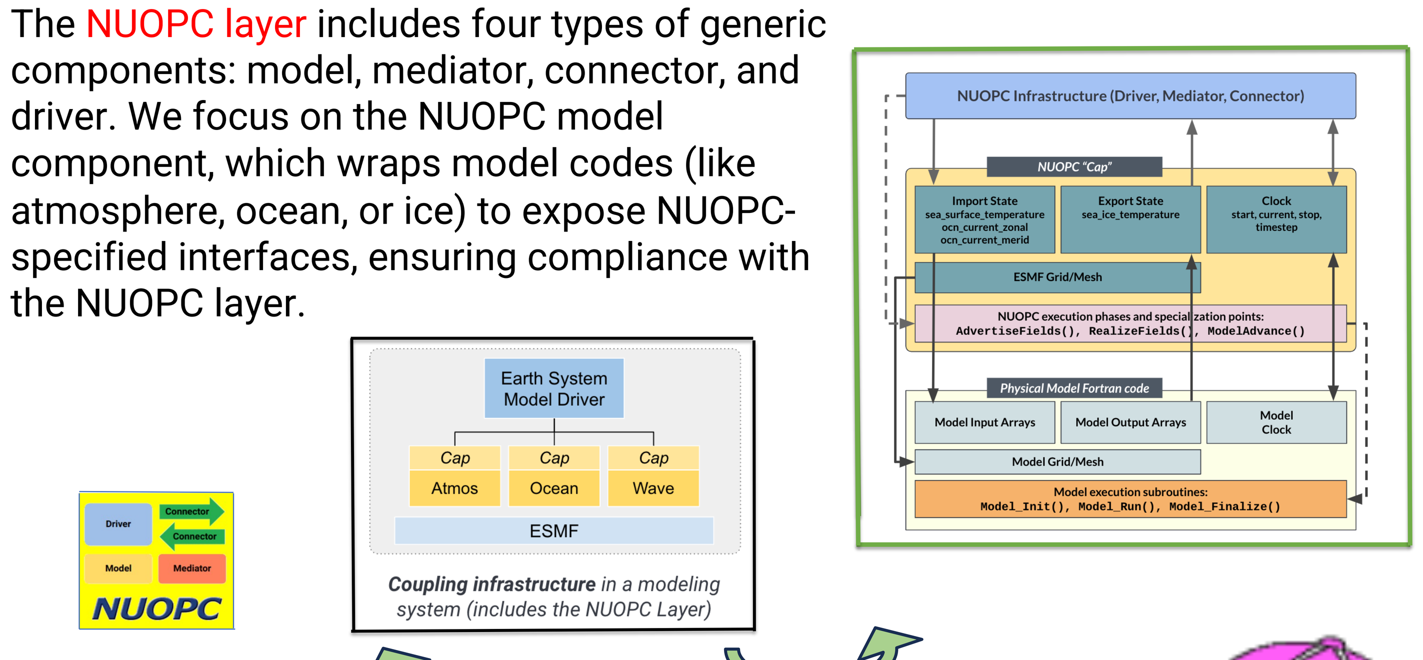
Currently, DA systems like NSF NCAR's Data Assimilation Research Testbed (DART) operate by writing restart files to disk, generating up to terabytes of data at every 6-hour timestep. We hypothesize that this I/O-intensive approach leads to significant performance bottlenecks.

NUOPC enables model components to share field data in memory, facilitating direct, in-memory exchanges. We integrated DART as a CESM model component using the NUOPC layer, which eliminates the I/O bottleneck by avoiding disk usage. This approach enhances performance and scalability, representing a significant advancement in Earth system modeling and data assimilation.

METHODS

Software Stack: ESMF (Earth System Modeling Framework) provides the essential utilities and libraries for building and integrating parts of Earth system models. NUOPC (National Unified Operational Prediction Capability) builds on ESMF, offering standards and guidelines to seamlessly connect these parts into a complete, operational model.

The NUOPC layer includes four types of generic components: model, mediator, connector, and driver. We focus on the NUOPC model component, which wraps model codes (like atmosphere, ocean, or ice) to expose NUOPC-specified interfaces, ensuring compliance with the NUOPC layer.

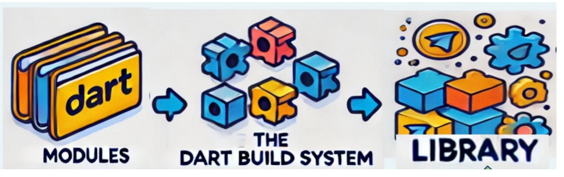


Utilizing NUOPC libraries to create a cap for DART software for the direct access of model state in memory.

Since DART is not a model but a software, we had to specialize the NUOPC cap to make it appear as a model to other model components.

Building DART as library

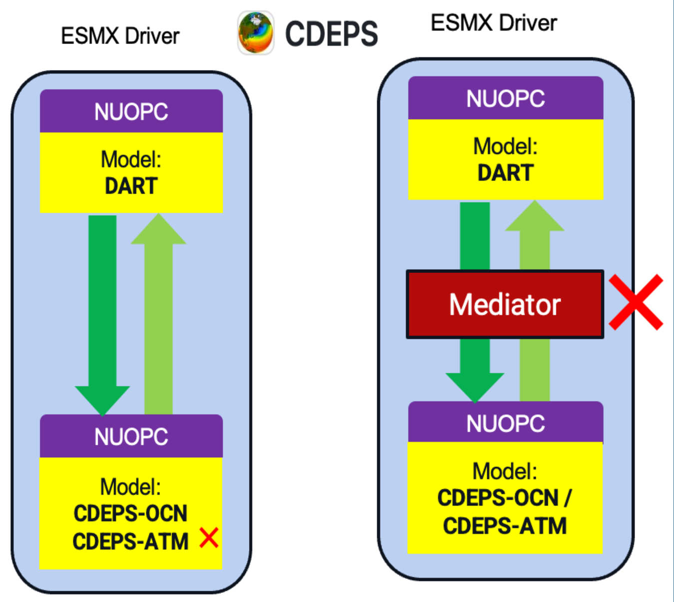
NUOPC wants the model component to build as a library and source code to be roughly divided into three execution method- initialization, run and finalize.



Structure of DART-NUOPC framework

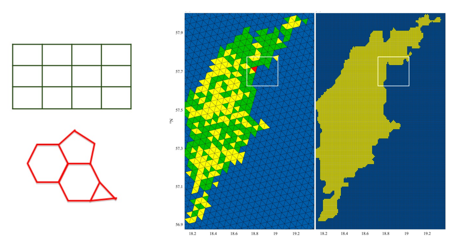
We started testing the DART-NUOPC cap code by coupling with a CDEPS data model component and went for ocean component for ease of use.

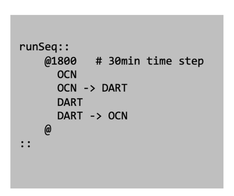
Unlike models, DART doesn't have a grid and therefore doesn't need the field data to be interpolated on certain grid, and therefore don't need mediator.



Field Mirroring in DART

Generalizing DART-NUOPC cap to accept all the field that the ocean model component must provide for DA and to get comfortable with both Mesh and Grid.





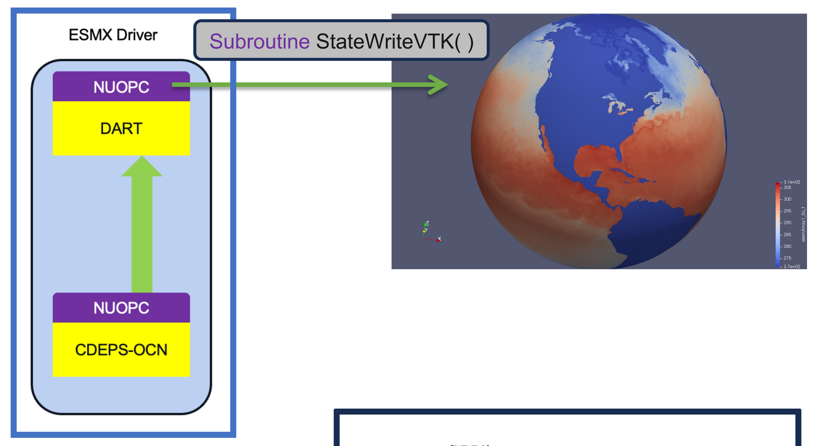
DART advancing in time

DART software doesn't have a clock which advances in time, and to make it appear as a model component we synchronized DART-NUOPC cap clock with the Driver's clock.

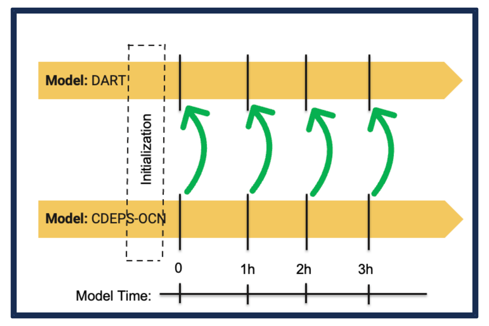
RESULT

We successfully developed NUOPC cap for DART, enabling efficient in-memory transfer of the 2-D Sea Surface Temperature (SST) field from OCN data model component to DART.

The Model Advance subroutine in NUOPC-DART cap correctly writes the accepted SST field to a VTK file, demonstrating that the field is transferred in memory.



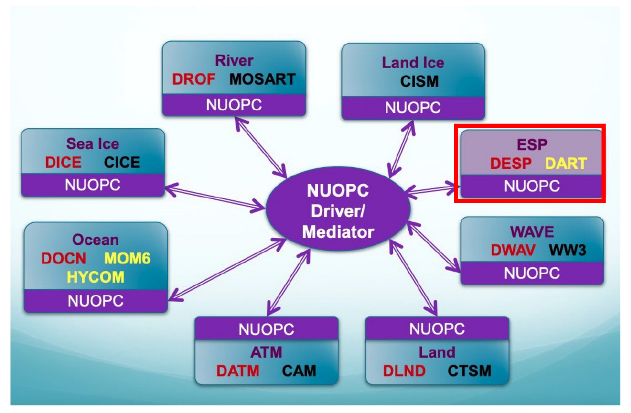
DART appears to advance in time as a model component, confirming the proper integration and functionality of the NUOPC cap. The following figure represent the data flow and time stepping of DART and CDEPS OCN data model component in a coupled environment.



FUTURE WORK

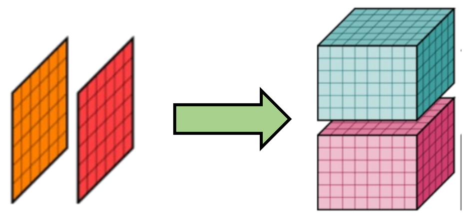
Integration with CESM

While we have developed an isolated testing environment using ESMX capabilities, our goal is to integrate this setup with the ESMF driver. This integration will enable us to leverage ESMF's comprehensive capabilities and apply our in-memory field transfer method within the full Earth system modeling framework.



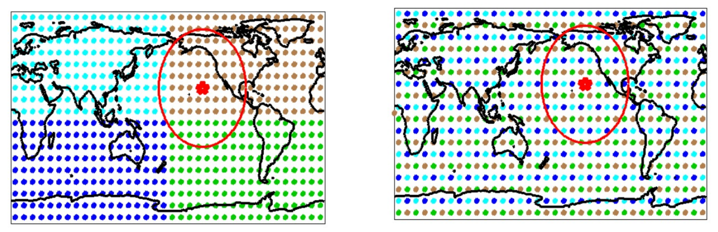
Performance Testing with 3D Fields

Currently, we have implemented and tested the in-memory transfer of 2D fields. Our next step is to extend this functionality to 3D fields. This will involve assessing the performance and scalability of 3D field transactions, which are more complex and data-intensive.



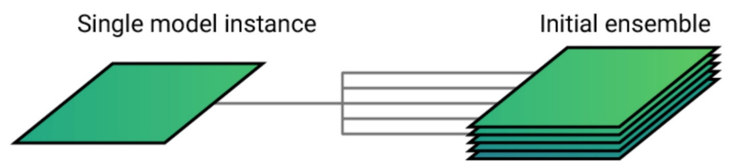
Optimizing Processor Distribution

At present, DART is distributed across the same number of processors as the CDEPS OCN data model component. To potentially enhance performance, we will experiment with round-robin distribution methods. This approach may offer better load balancing and resource utilization, leading to improved efficiency.



Ensemble Field Transfers

Our current implementation transfers a single field. Future work will focus on enabling the transfer of ensemble fields, which are crucial for robust data assimilation processes.



ACKNOWLEDGMENTS

Thanks to mentors Dan Amrhein, Helen Kershaw and Ufuk Turuncoglu, for consistent support throughout the project, program coordinators, Virginia Do, Jerry Cyccone for making this internship a seamless experience and fellow interns for an amazing summer. Special thanks to ESMF and DAReS group for teaching DART concepts and ESMF software. Also, thanks to the NCAR, CISL Help Desk and ASAP team for providing support during the internship.