Optimizing Ensemble Data Assimilation Performance for Coupled Earth System Models

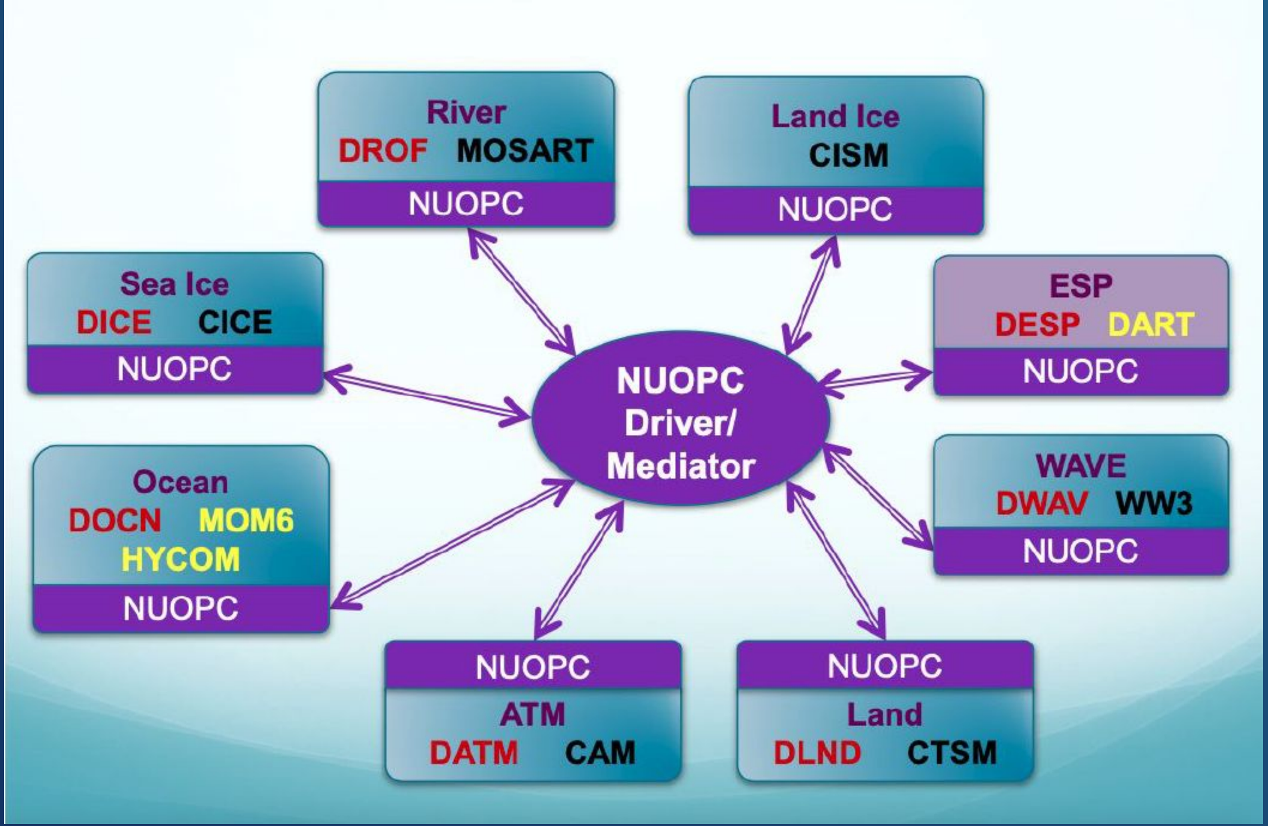
A first prototype for in-memory data transfer between earth system models and Data assimilation

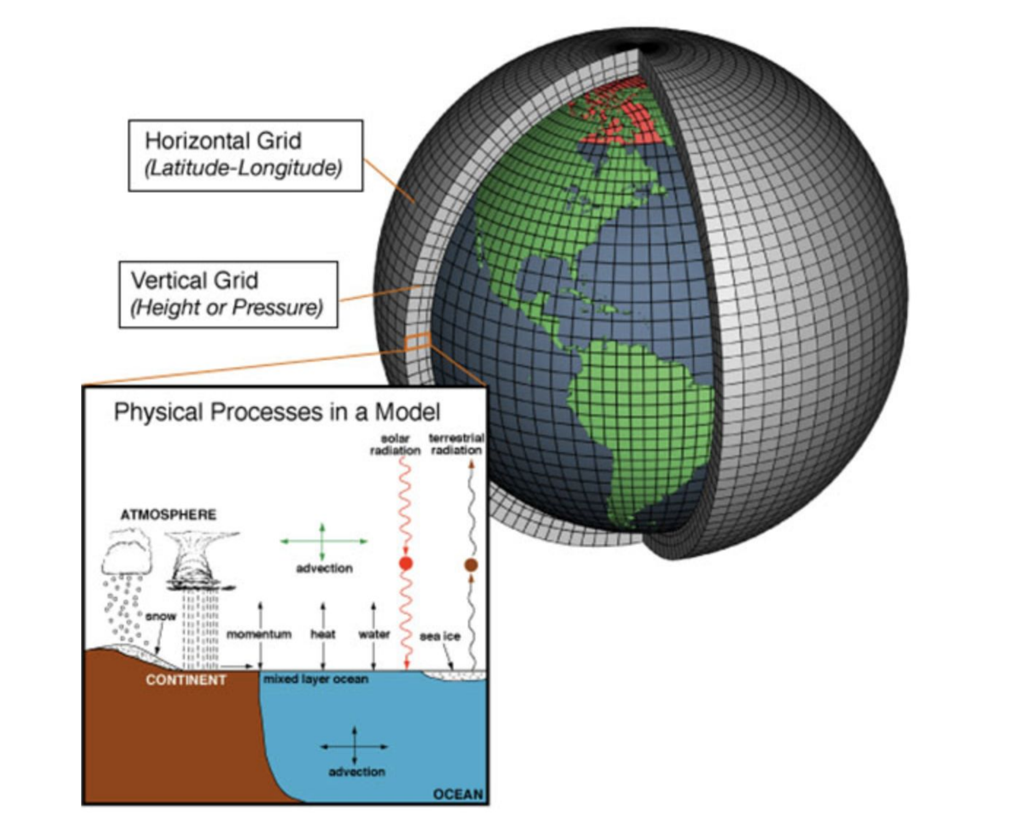
Suman Shekhar, *Rutgers University, New Jersey*
Anh Pham, Dan Amrhein, Helen Kershaw, Ufuk Turuncoglu, Dan Rosen

This material is based upon work supported by the National Center for Atmospheric Research, which is a major facility sponsored by the National Science Foundation under Cooperative Agreement No. 1852977.

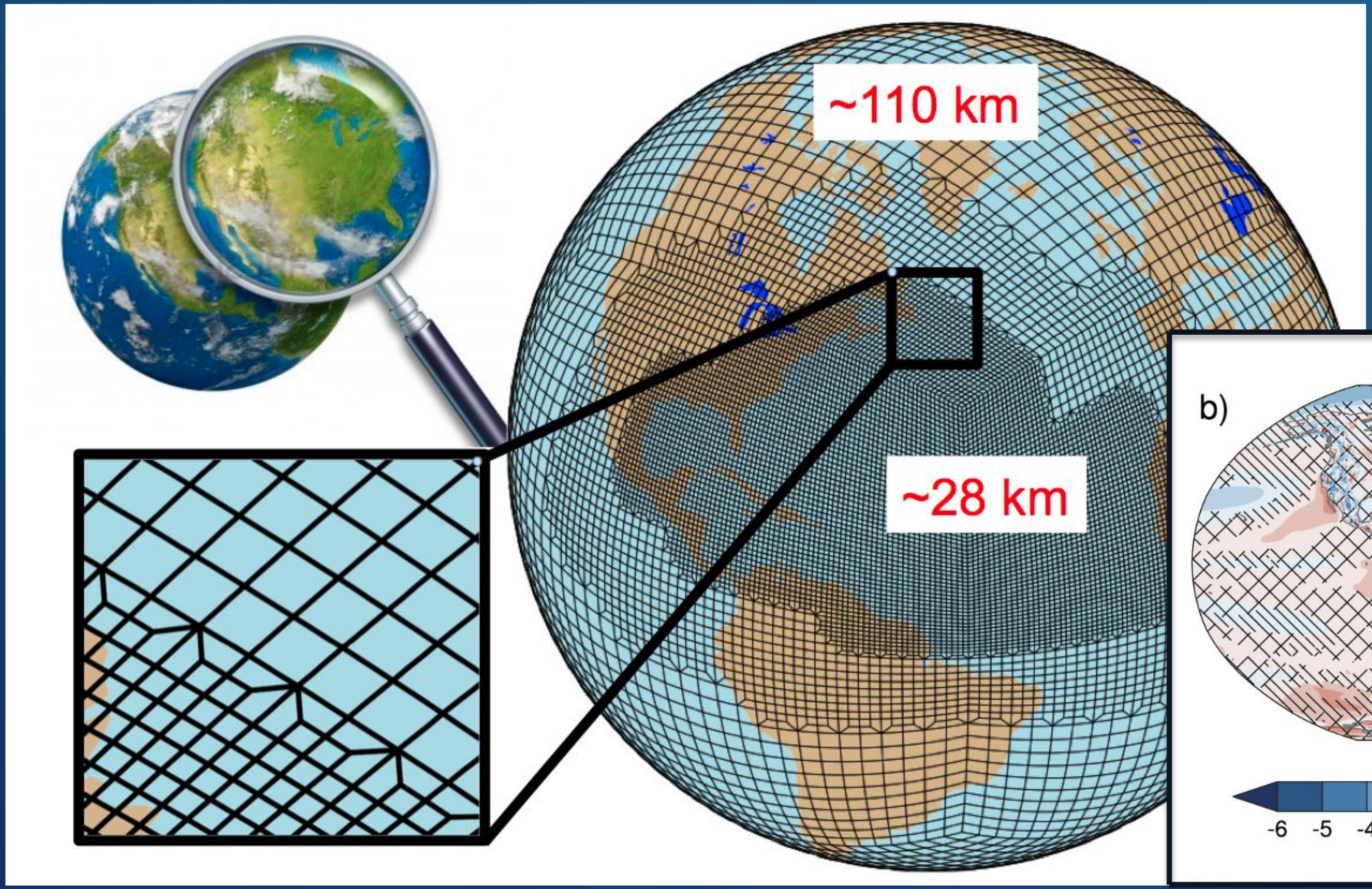
- Introduction
- Method: Building the DART cap
- Results
- Future work

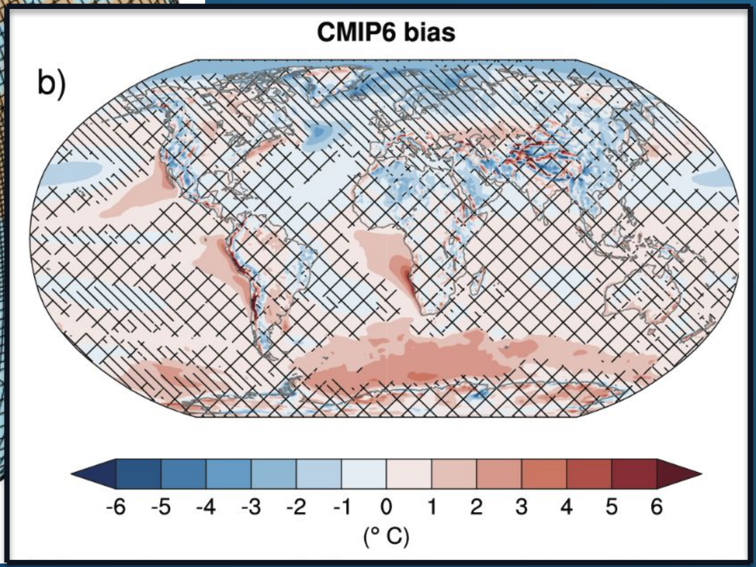
Community Earth System Model (CESM)

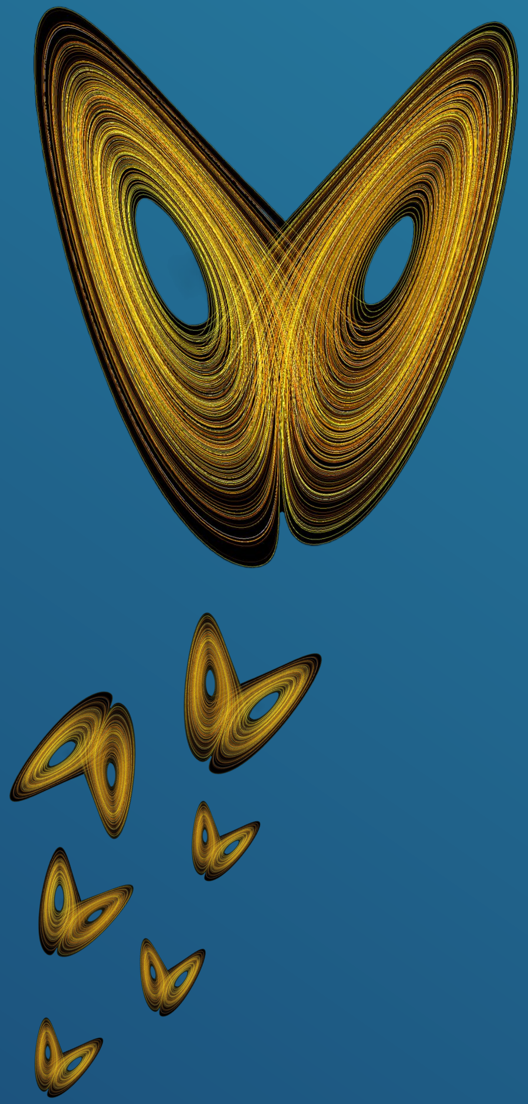




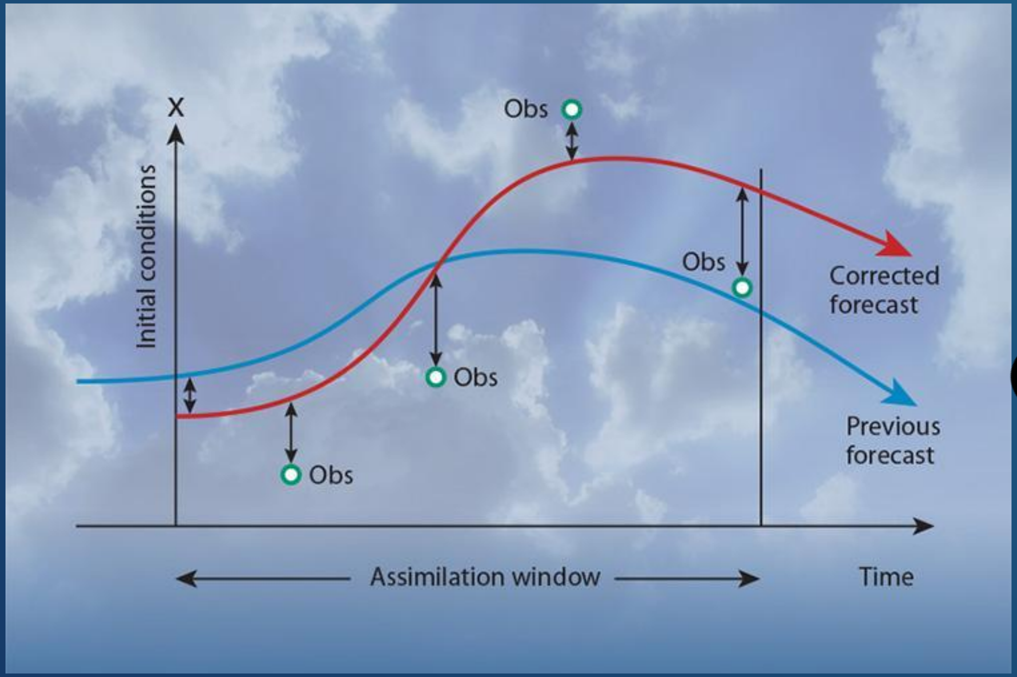
Climate Change is a complex problem!

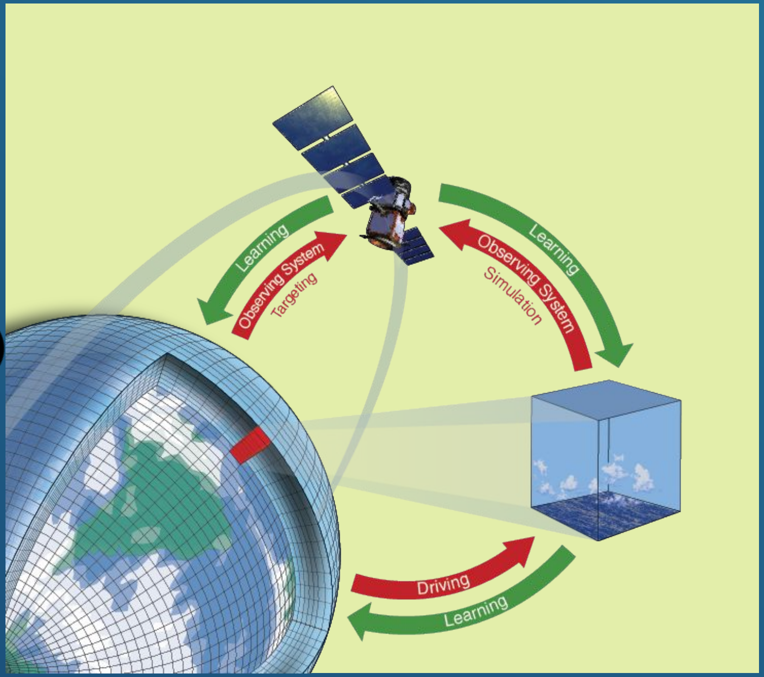






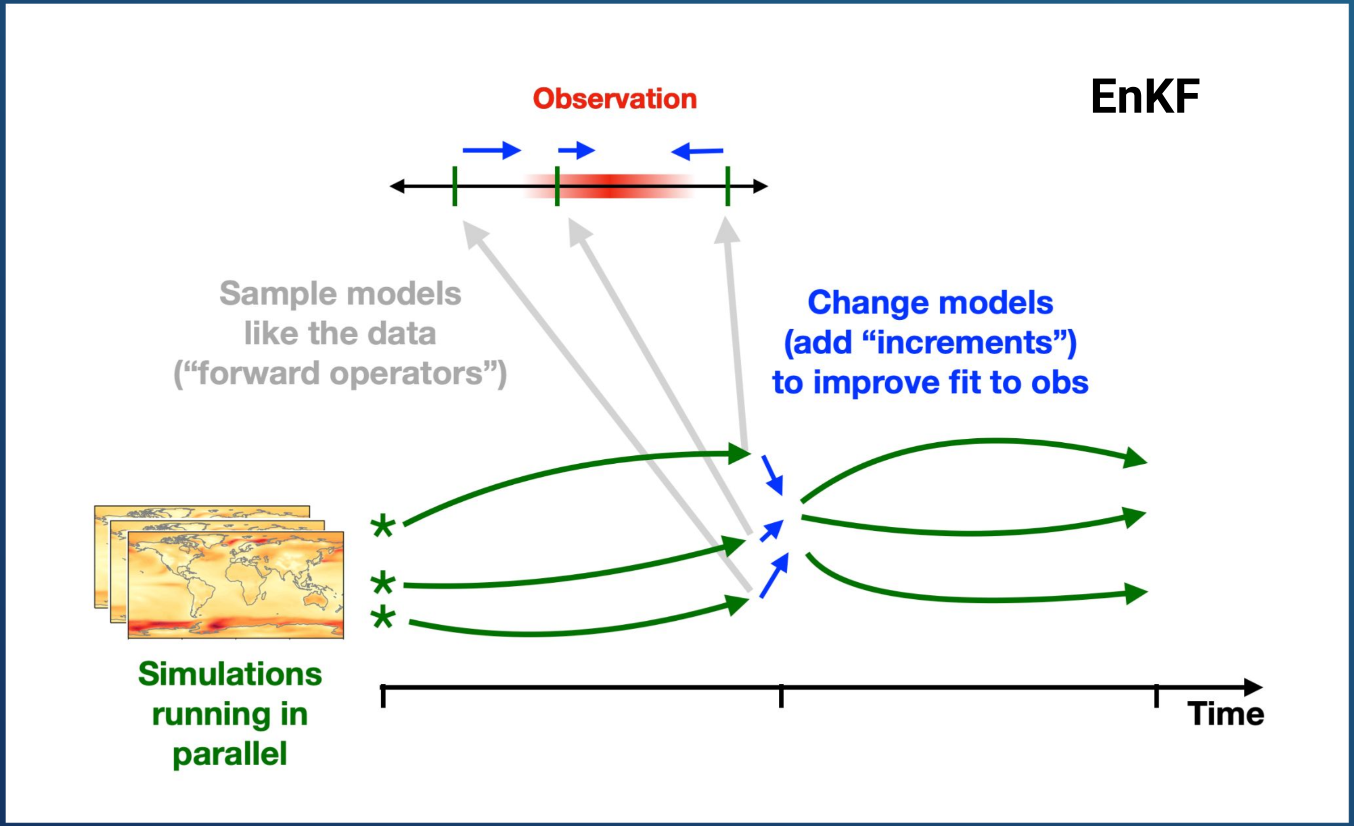
We need both data and algorithms to connect the dots

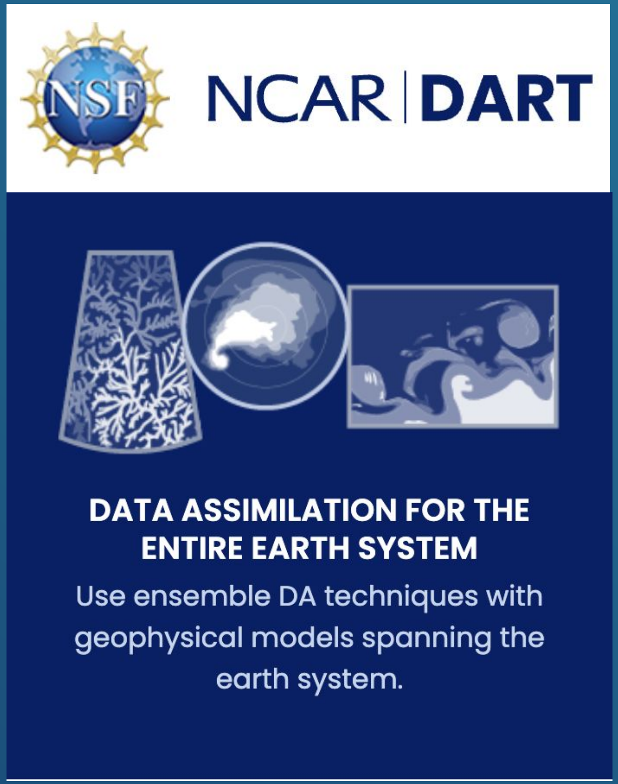




Algorithm: Data assimilation combines observations with model forecasts to estimate the state of a physical system

DA in a nutshell





Challenge: I/O bottleneck of models -> DART -> models

Solution: In-memory transfer of fields leveraging NUOPC software layer

- Coupled models like CESM already exchange fields in memory using Earth System Modeling Framework (ESMF) utilities.
- We are proposing to integrate DART software as a model component within CESM and use NUOPC libraries to create a cap for DART software to access the model state in memory.

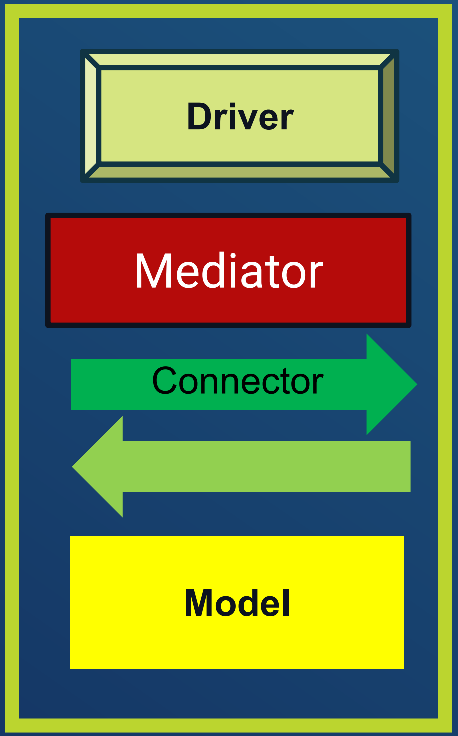
- Introduction
- Method: Building the DART cap
- Results
- Future work

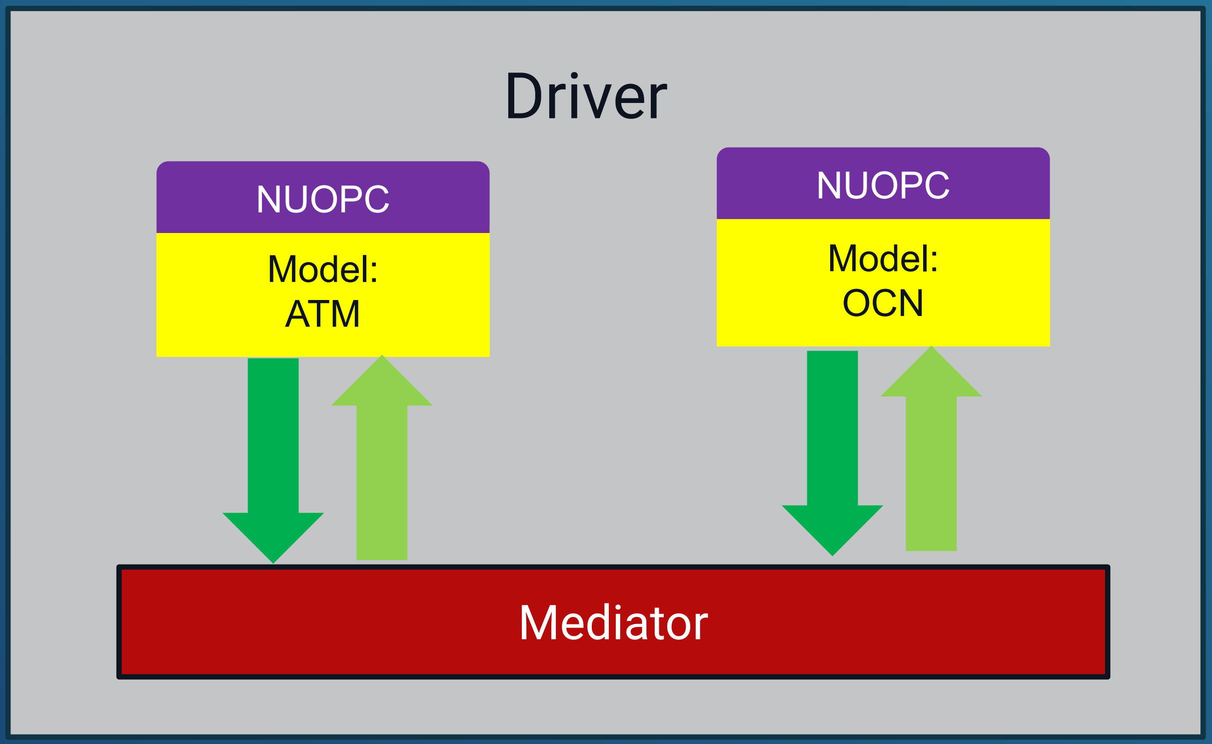
Difference between ESMF and NUOPC

- Software Stack: ESMF (Earth System Modeling Framework) provides 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 connect components into a complete, operational model.

Example of coupling atmospheric model and ocean model





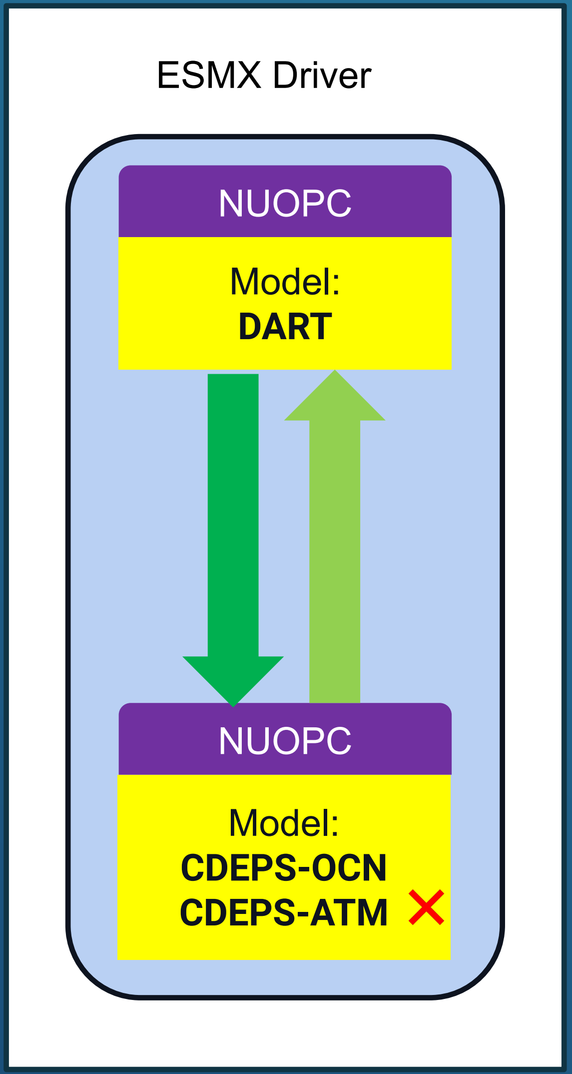
The NUOPC layer includes four types of generic components: model, mediator, connector, and driver. We focus on the NUOPC model component, which wraps model codes (like atmosphere, ocean, or ice) to expose NUOPC-specified interfaces, ensuring compliance with the NUOPC layer.

Structure of DART-NUOPC Framework

- We started testing the DART-NUOPC cap code by coupling with a CDEPS (Community Data Models for Earth Prediction Systems) data model component and went for ocean component for ease of use (and also because the ocean is the best component).

- 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 and the following points we had to take care of -

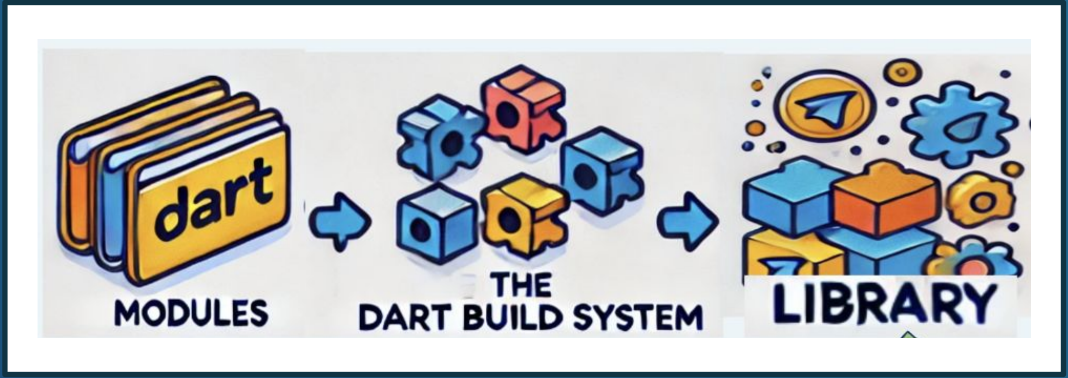
  1. DART builds as an executable.
  2. DART doesn’t need any operation to be done on State Variable.
  3. DART doesn’t have its own Grid/Mesh.
  4. DART doesn’t step forward in time.



CDEPS

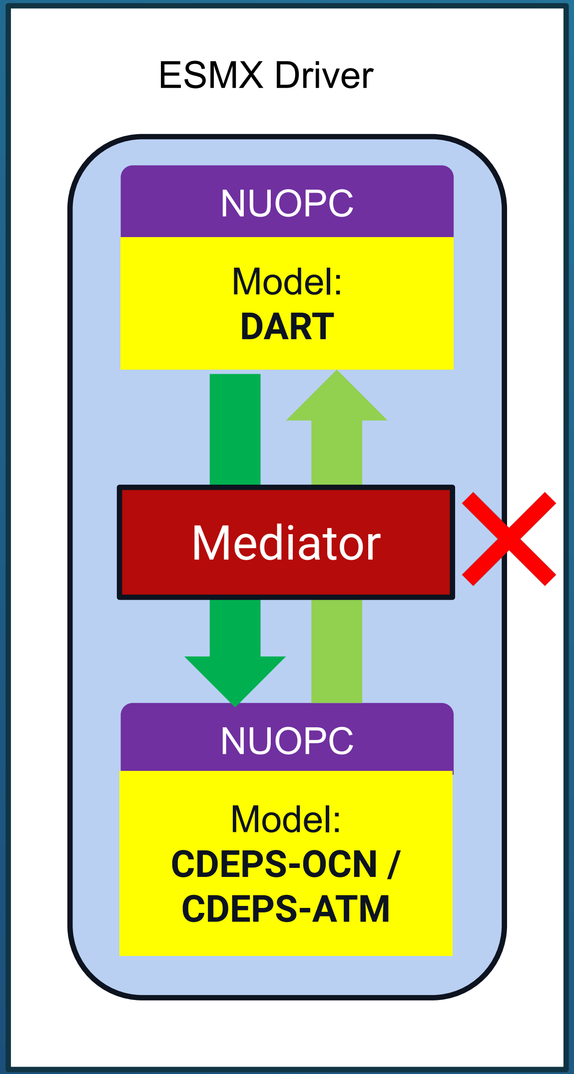
1. Building DART as library

- DART is already an independent component.
- NUOPC wants your model to built as a library.
- DART needs to be roughly divided into several execution methods: ***initialize, run, and finalize***.



2. No need of Mediator in case of DART

- Unlike models, DART doesn't want to do any custom operations on the state variable, and therefore don't need mediator component.



3. Field Mirroring & Receiving Grid/Mesh

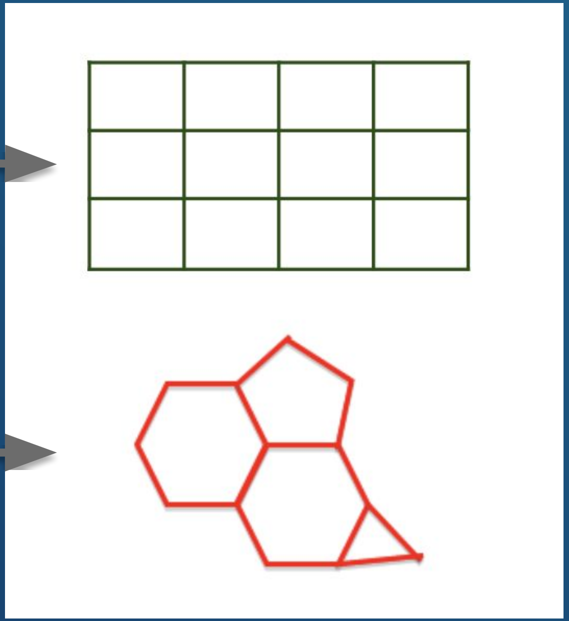
- Generalizing DART-NUOPC cap to accept all the field that the ocean model component must provide for DA and to get comfortable with both Mesh and Grid

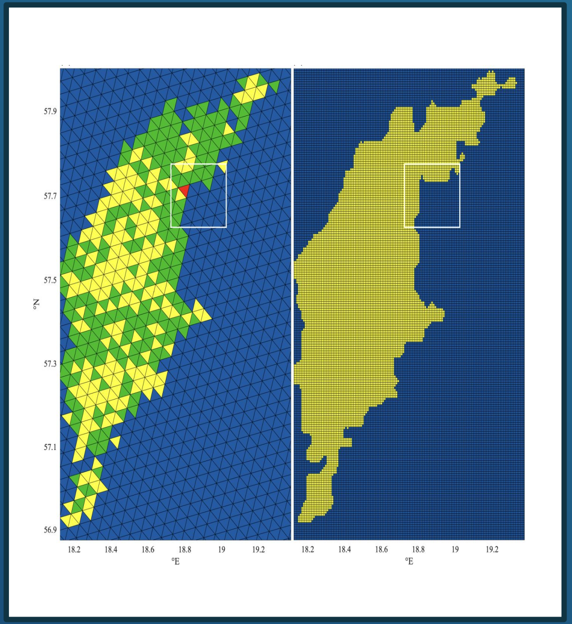
Grid

A **structured** representation of a region, such as a logically rectangular tile or set of tiles

Mesh

An **unstructured** representation of a region including 2D polygons with any number of sides and 3D tetrahedra and hexahedra





4. DART Time Stepping

- 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.

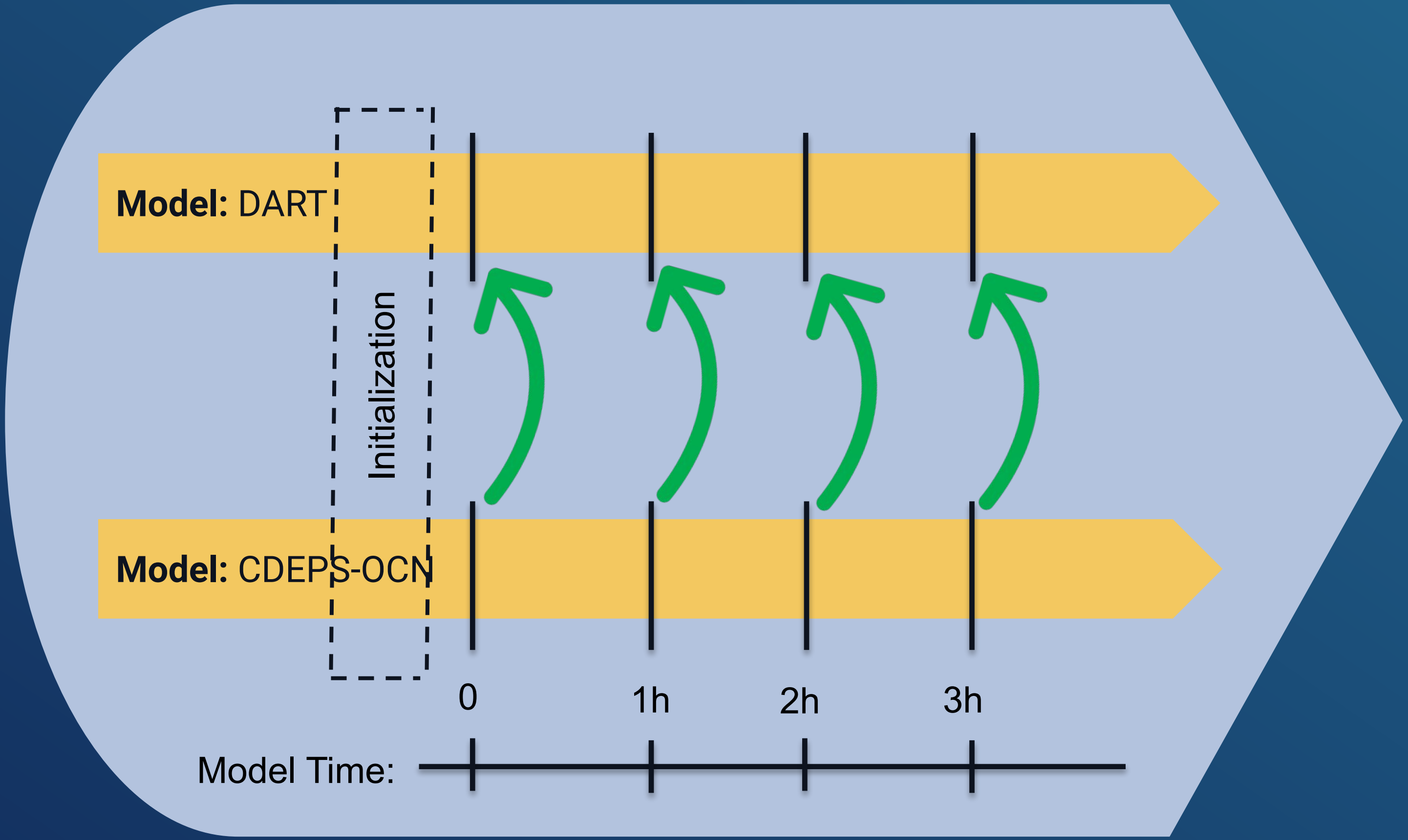
```
runSeq::
    @1800   # 30min time step
      OCN
      OCN -> DART
      DART
      DART -> OCN
    @
::
```

A driver with two model components (DART and OCN), and connectors.

Driver reads a run sequence from a yaml file

- Introduction
- Method: Building of DART cap
- Results
- Future work

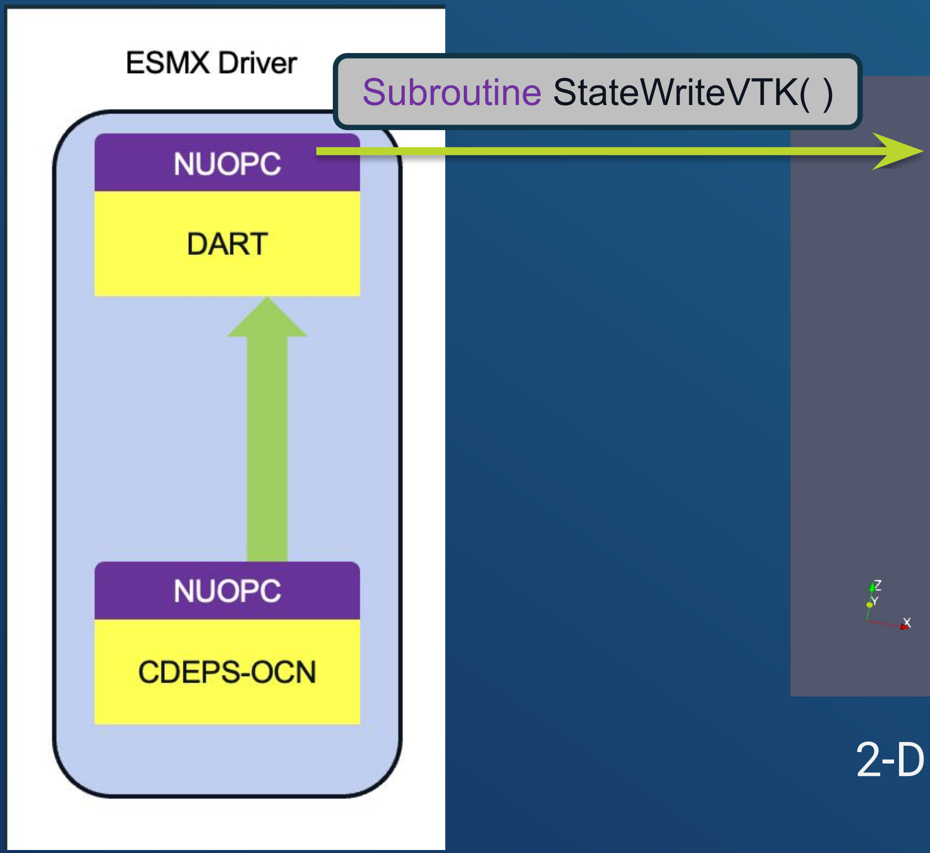
Coupling DART with OCN component

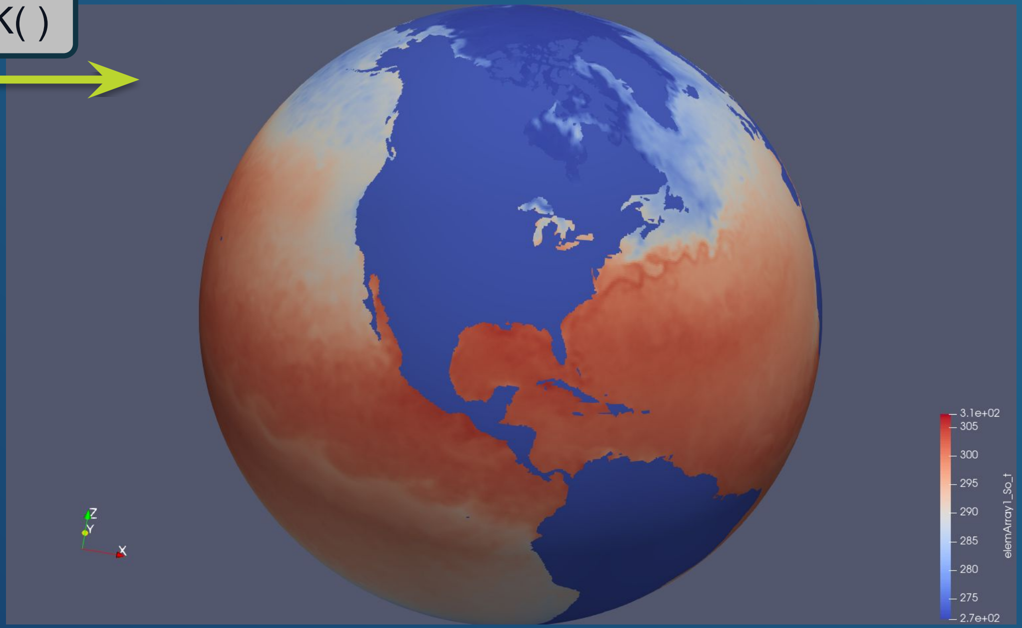


- DART appears to advance in time as a model component, confirming the proper integration and functionality of the NUOPC cap.
- The following figure represent the data flow and time stepping of DART and CDEPS OCN data model component in a coupled environment.

Result: DART model component has successfully accepted the field in-memory!

- 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.



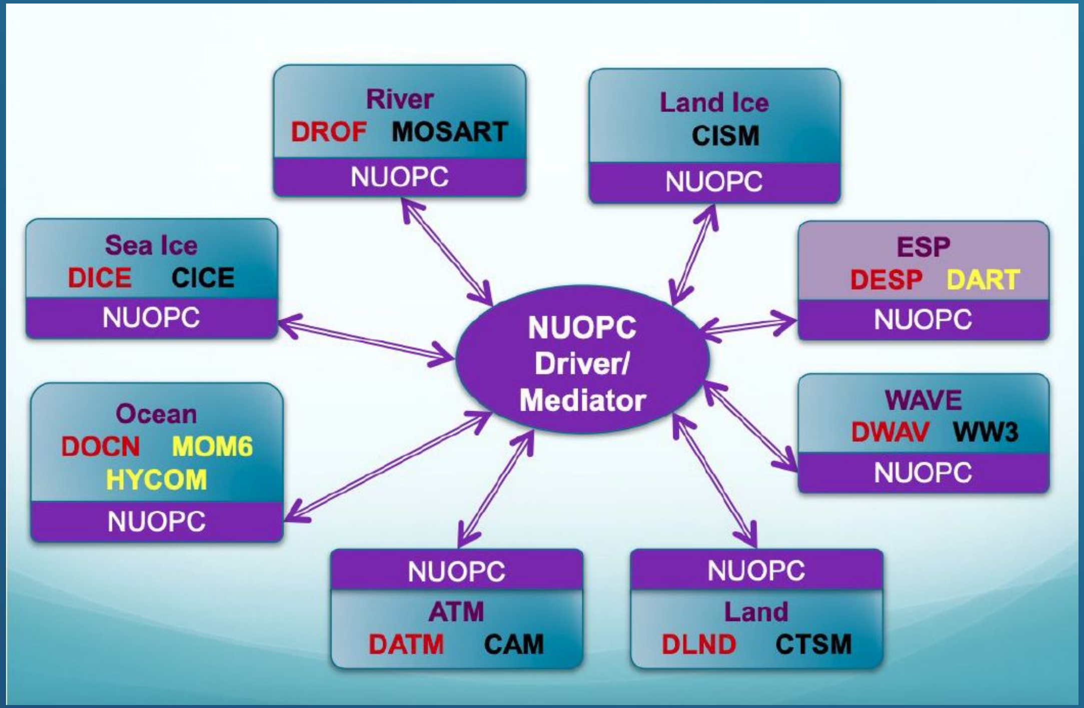


2-D Sea Surface Temperature (SST) field received by DART

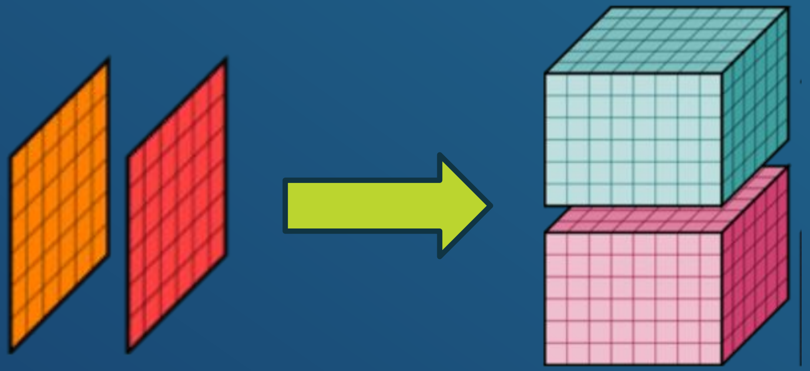
- Introduction
- Method: Building of DART cap
- Results
- Future Work

Future Work

- Integration with CESM

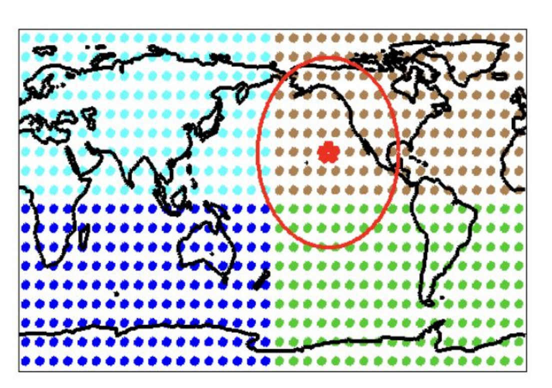


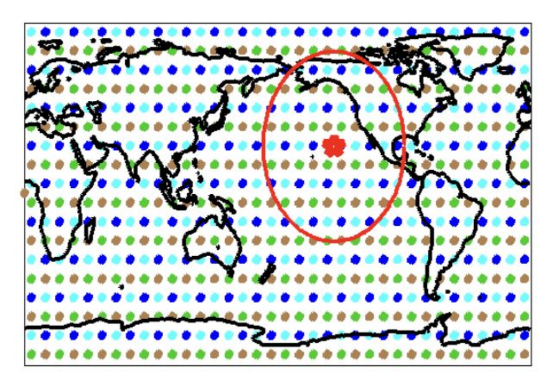
- Performance Testing with 3D Fields



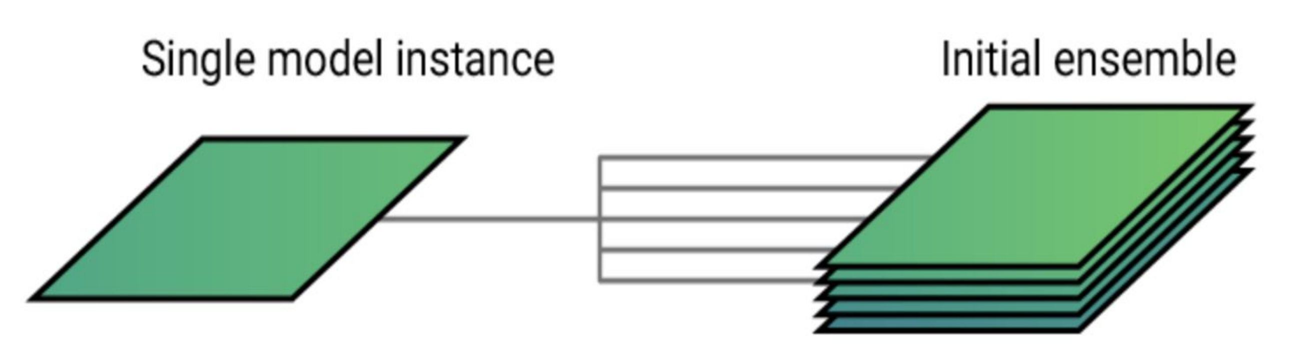
Future Work

- Optimizing Processor Distribution

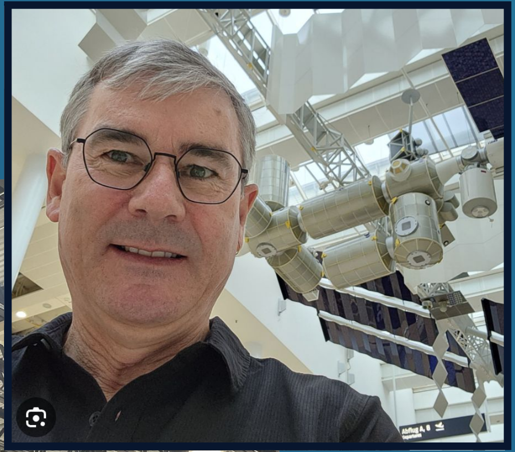




- Ensemble Field Transfers



Thank You!



Summer Internships in Parallel Computational Science
SIParCS
BOULDER · CO

NSF | NCAR
Operated by UCAR

ESMF

Reference Slides

NUOPC compliant Data model component

CDEPS contains ESMF/NUOPC compliant data components that are modular and flexible: Can be used in any ESMF/NUOPC compliant modeling system

Main building block is the data stream.

○ All fields in a stream are located in the same data file/s

○ All share same spatial and temporal properties

○ Data models can have multiple streams: e.g., SST data could originate from OISST and precipitation data could come from CRU.

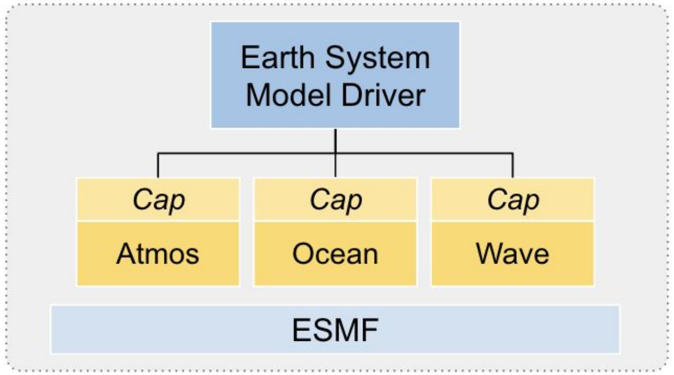
● All data is read with parallel IO (PIO2)

Framework for Coupled Modeling

The Earth System Modeling Framework (ESMF) is high-performance software infrastructure used in coupled Earth science applications.

Key Features:

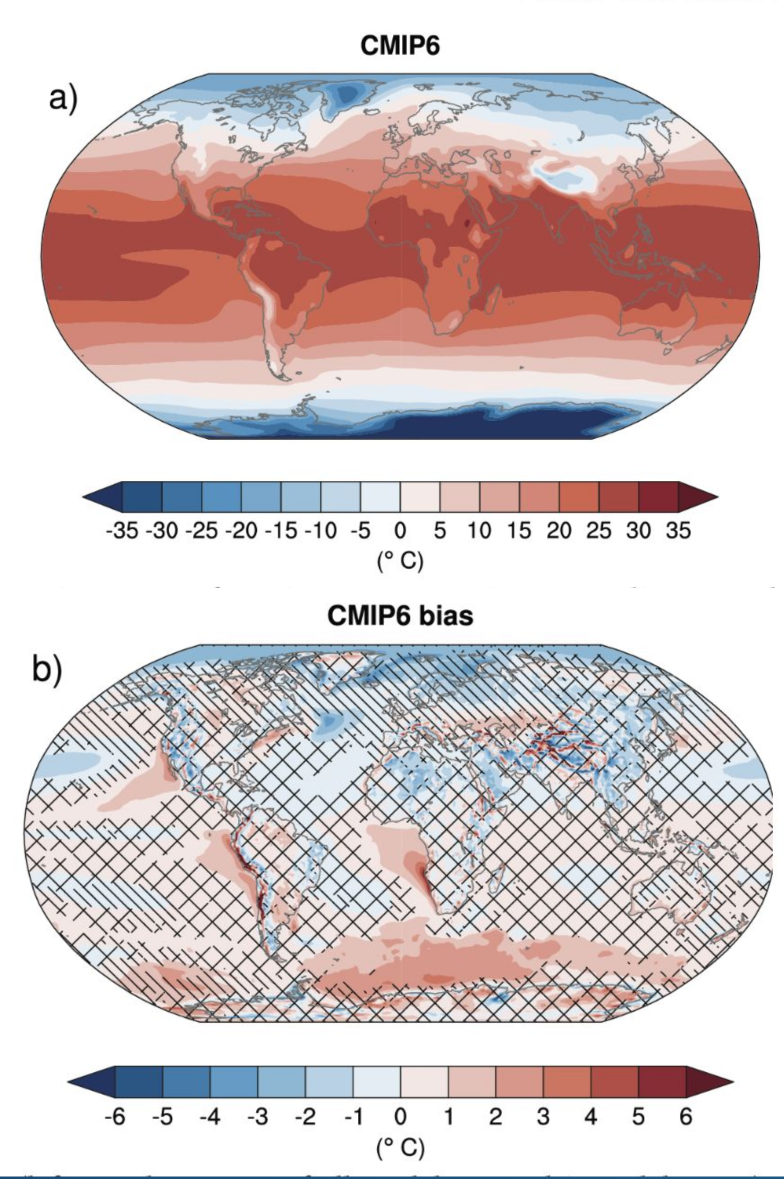
- Standardized Interfaces
- High-Performance Coupling
- Modularity and Reusability
- Support for Parallelism
- Interoperability



Coupling infrastructure *in a modeling system (includes the NUOPC Layer)*

Model Biases!

- Climate Model Large Ensembles Unable to Reproduce Observed Trends
- Discrepancies between simulated and observed fields are commonly referred to as biases
- All models have a bias. Bias occurs because models are a mathematical representation of a highly complex system which is a simplification of the reality of many processes. Models can have bias because of limiting spatial resolution or incomplete knowledge of how a process works.



I/O bottleneck: Models -> DART -> Models

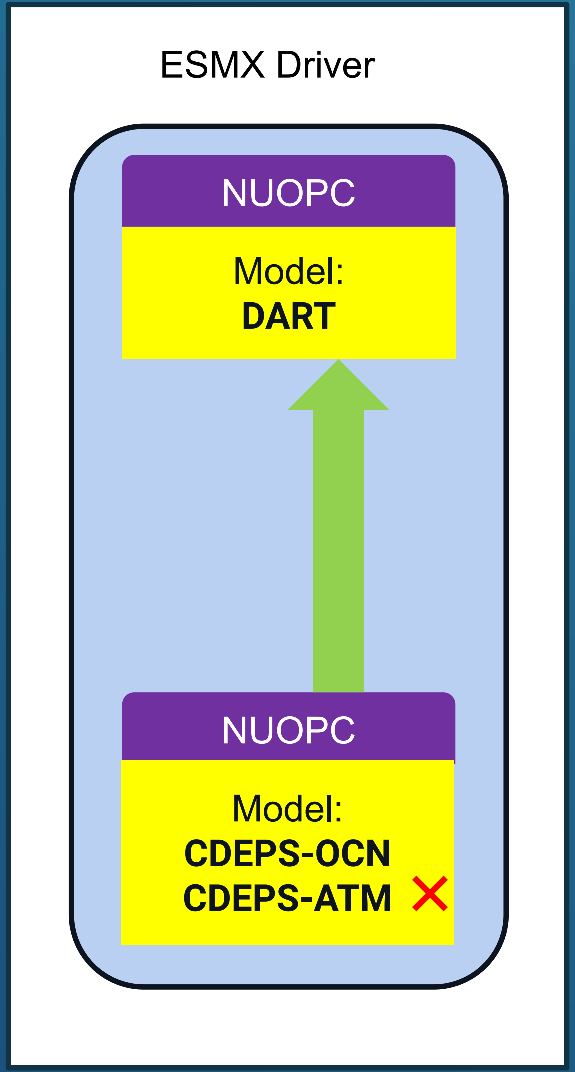
Current Approach:

- File-Based Data Transfer:
  - DART modifies "**restart**" files written to **disk**.
  - Frequent writing and reading of files introduce significant I/O overhead.
  - Disk I/O operations are inherently slower than in-memory operations.
- Model Execution Disruptions:
  - Model needs to stop for file write/read operations.
  - Stopping and restarting the model reduces overall computational efficiency.

Specializing DART Generic Model Component

ESMX_DARTOcnProto > NUOPC_DART-code-development > DART > models > cdeps > work

```fortran
module dart_comp_nuopc
    subroutine SetServices(dgcomp, rc) ⋯
    end subroutine SetServices


    subroutine InitializeAdvertise(dgcomp, rc) ⋯
    end subroutine InitializeAdvertise

    subroutine ModifyAdvertise(dgcomp, rc) ⋯
    end subroutine ModifyAdvertise

    subroutine RealizeAccepted(dgcomp, rc) ⋯
    end subroutine RealizeAccepted

    subroutine SetClock(dgcomp, rc) ⋯
    end subroutine SetClock

    subroutine ModelAdvance(dgcomp, rc) ⋯
    end subroutine ModelAdvance

    subroutine StateWriteVTK(state, prefix, rc) ⋯
    end subroutine StateWriteVTK


end module dart_comp_nuopc
```



Recap: NUOPC generic component layers

- A NUOPC component is an ESMF component with specified rules of behavior depending on the component's role in the coupled system.

Driver

Driver

Connector

Connector

Model

Model

Mediator

Mediator