Optimizing Ensemble Data Assimilation for Coupled Earth System Models

DART-X: Software Infrastructure for Prototyping in-memory Data Transfer between Ensemble Data Assimilation and Coupled Earth Systems Models

Anh Pham,
Suman Shekhar, Helen Kershaw, Dan Amrhein, Ufuk Turuncoglu

***N**ational **C**enter for **A**tmospheric **R**esearch (**NCAR**)*
***S**ummer **I**nternships in **P**arallel **C**omputational **S**cience (**SIParCS**)*

JULY 30, 2024

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

Outline

1. Introduction Background

- *Climate Modeling,* ***CESM***
- ***Data Assimilation***
- ***DART*** *Software*

2. Motivation Project Goals

- ***Profiling*** *results that shows* ***I/O bottlenecks***
- *Problem statement*
- *Why the* ***'Cap'*** *(****interface*** *for* ***CESM and DART****)?*

3. Methods Results

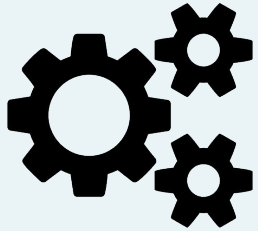
- ***Infrastructure*** *Challenges:*
  - *Derecho (HPC) vs. Docker (container)*
  - *ESMF, NUOPC, ESMX*
- ***ESMX*** *Framework decision*
- *The* ***Build Process*** *to* ***integrate*** *software, drivers and models for* ***Data Assimilation of Coupled Models***

4. Conclusions Future directions

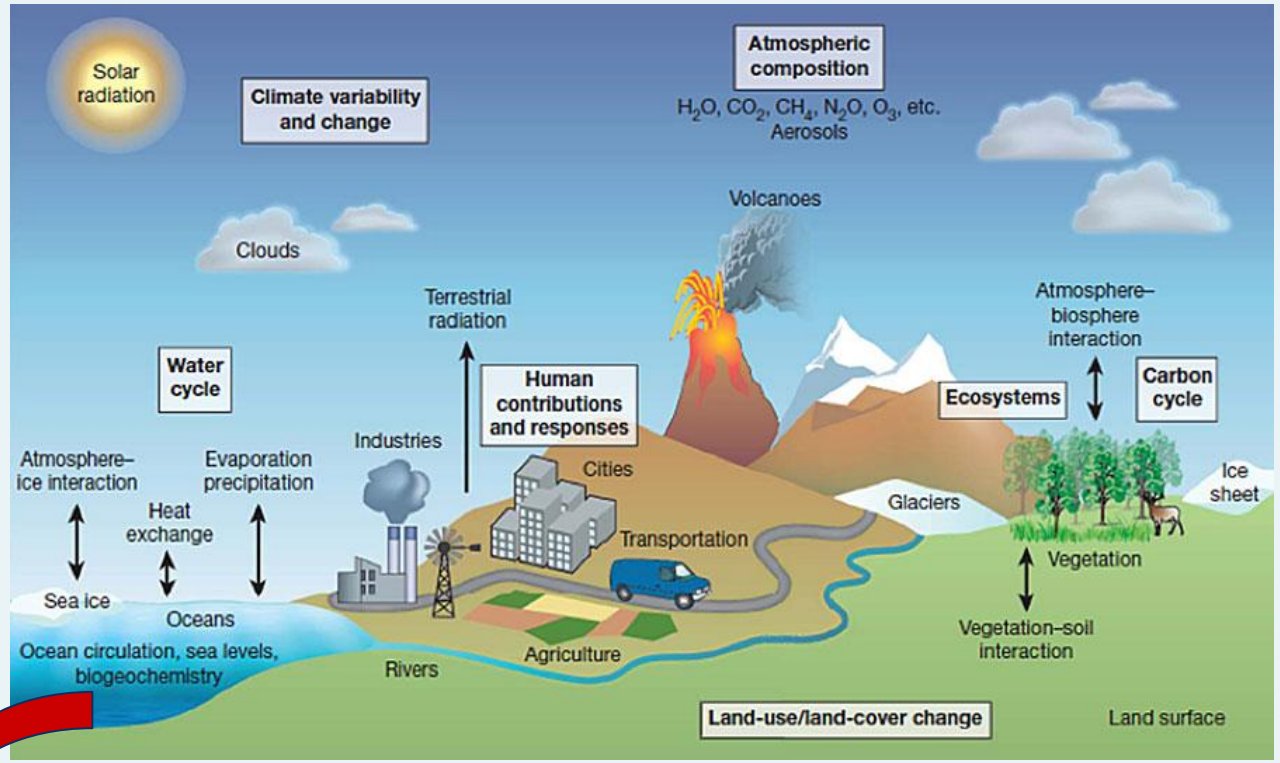
- *Advancing Climate Sciences:* ***Computational Frameworks and Software Infrastructure***
- *Direct Data Sharing in* ***Memory***
- ***Validate/Confirm*** *Results:* ***Profiling Cap Performance***

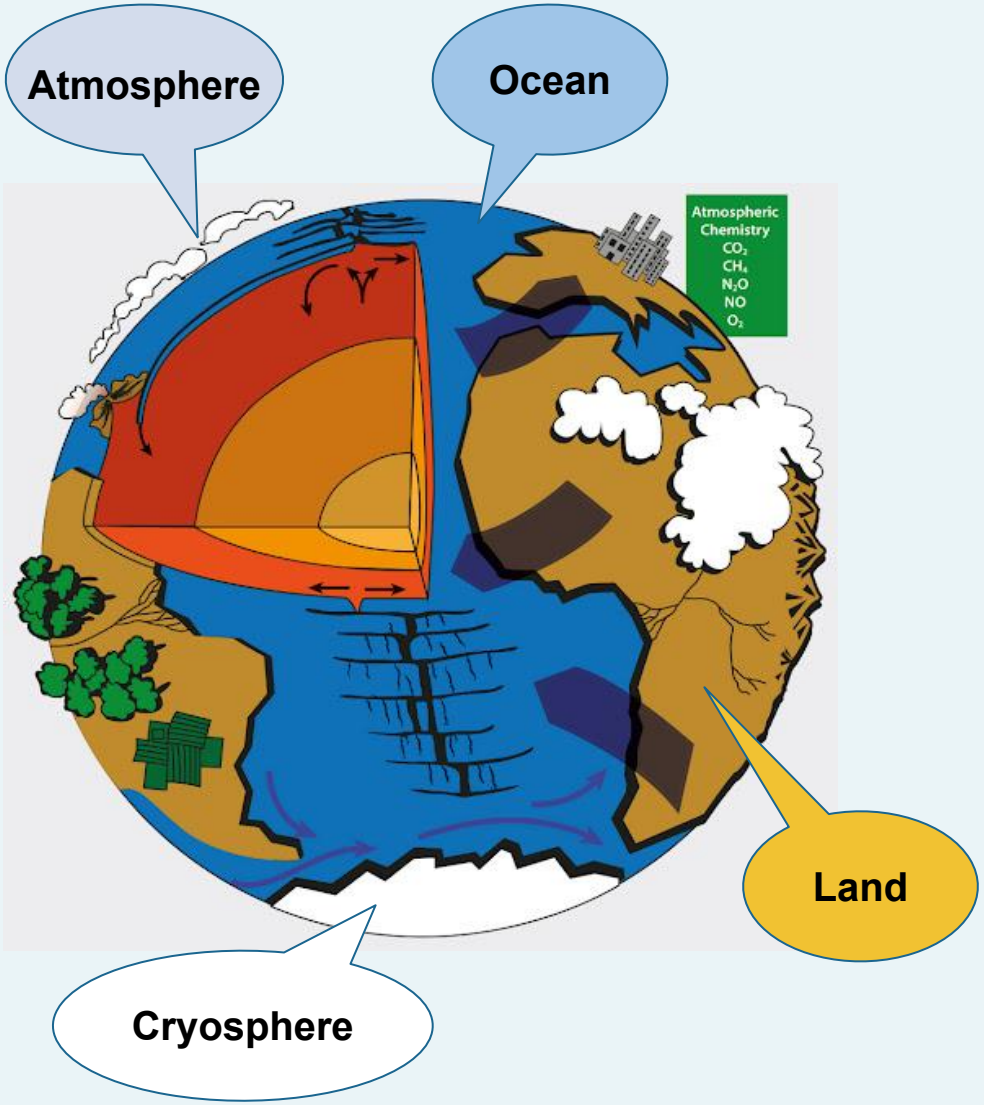
Background: Essential applications of Earth System predictions

- Reliable climate predictions have **powerful applications**:
  - ➢ Daily safety and Convenience
  - ➢ Policy and Decision-Making
  - ➢ Business and Economy



Climate Modeling Challenge

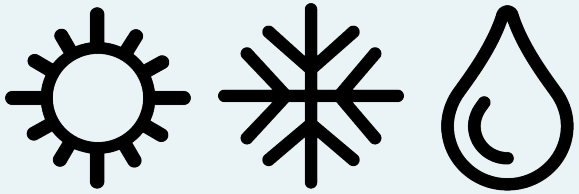




Components do not act in isolation!

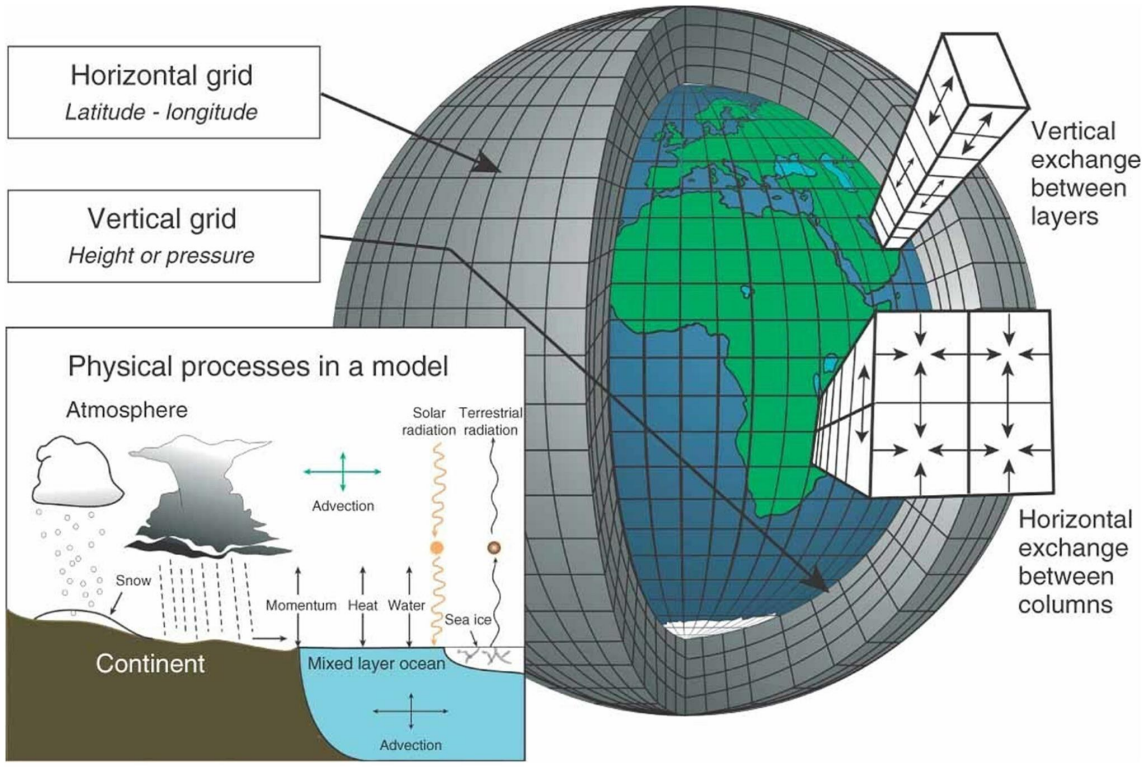
Earth System Modeling

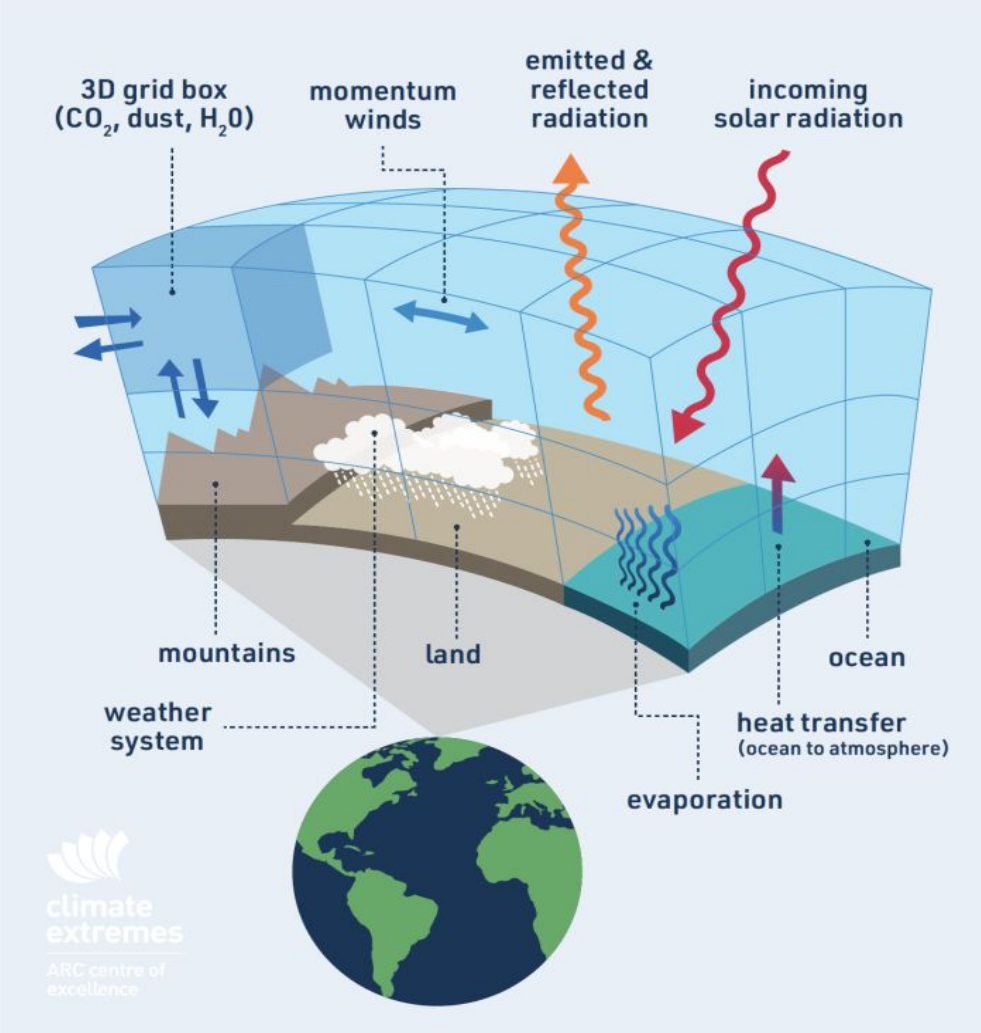
Earth System Modeling
enhances climate predictions



- Treats the Earth as an **integrated system** (as it is)
- **Interactive components** that made up the Earth system
- **C**ommunity **E**arth System **M**odel (**CESM**)

Coupled Climate Models

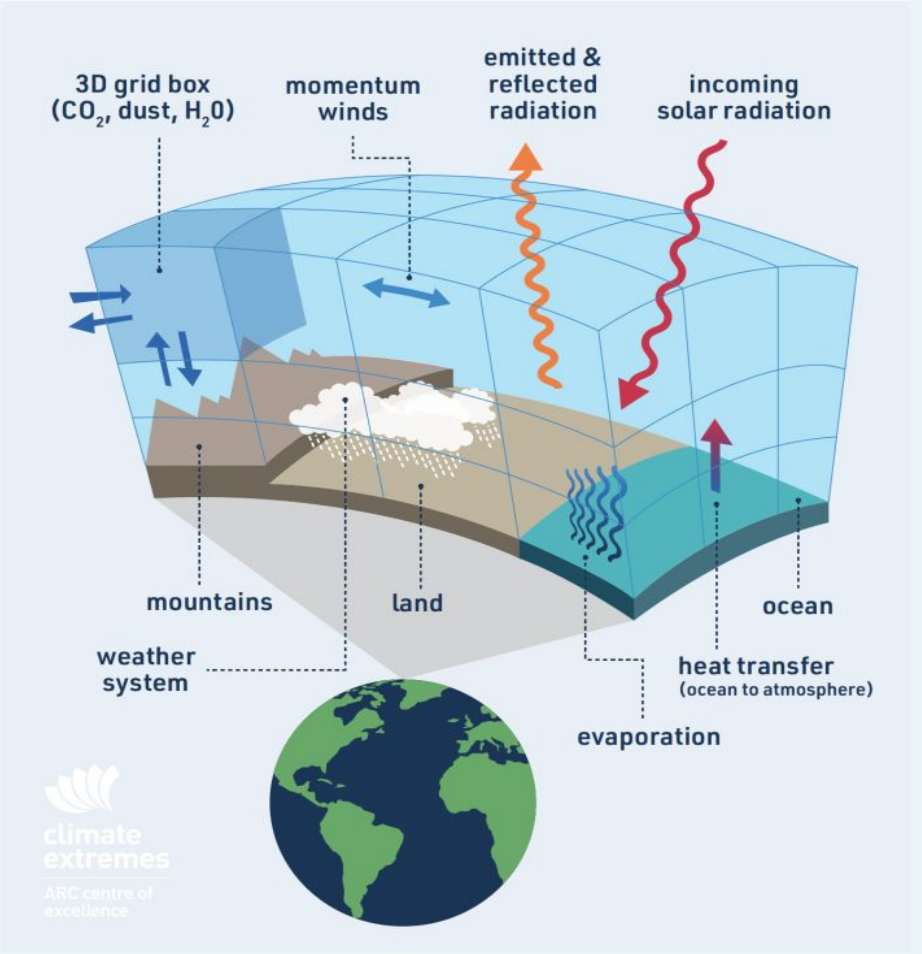




Earth System Modeling

Sounds awesome, too good to be true?

- Modeling complex climate processes is …**complex!**
  - ➢ High-dimensionality
  - ➢ Biased models
  - ➢ Expensive to run and re-run

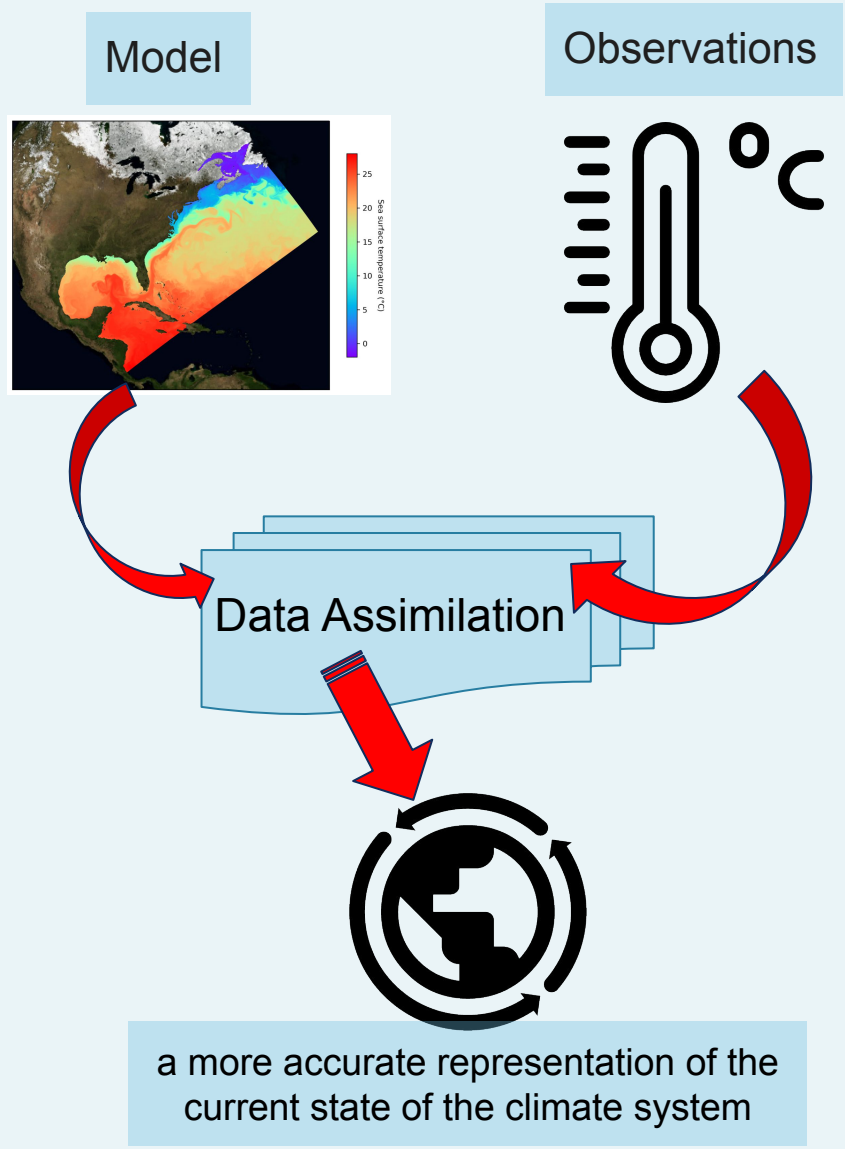


Data Assimilation
DART Software

What is Data Assimilation (DA) ?

- Both computer models and observational data have **uncertainties**:
  - ➢ Models: can be oversimplified and biased
  - ➢ Data: can have errors due to limitations in measurement instruments and coverage

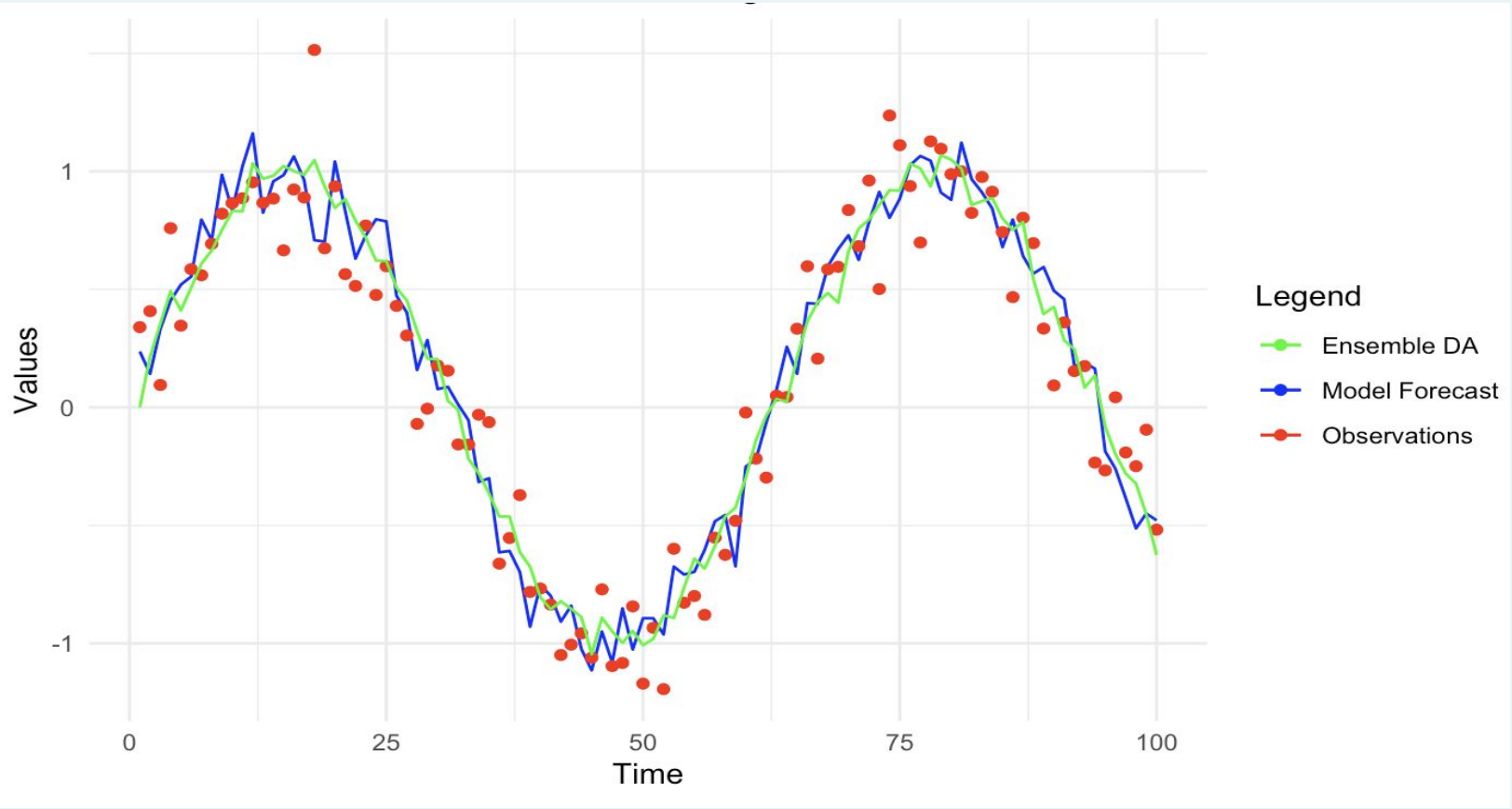
- **Data Assimilation** is a technique to combine **computer models** and **observational data** to balance out uncertainties.



What is DART?

DART Software: Data Assimilation Research Testbed

Pulling Observations Closer to the Model



Project Title Review

DART-X: Software Infrastructure for Prototyping **in-memory** Data Transfer between Ensemble Data Assimilation and Coupled Earth Systems Models

Outline

1. Introduction Background

- *Climate Modeling,* ***CESM***
- ***Data Assimilation***
- ***DART*** *Software*

2. Motivation Project Goals

- ***Profiling*** *results that shows* ***I/O bottlenecks***
- *Problem statement*
- *Why the* ***'Cap'*** *(****interface*** *for* ***CESM and DART****)?*

3. Methods Results

- ***Infrastructure*** *Challenges:*
  - *Derecho (HPC) vs. Docker (container)*
  - *ESMF, NUOPC, ESMX*
- ***ESMX*** *Framework decision*
- *The* ***Build Process*** *to* ***integrate*** *software, drivers and models for* ***Data Assimilation of Coupled Models***

4. Conclusions Future directions

- *Advancing Climate Sciences:* ***Computational Frameworks and Software Infrastructure***
- *Direct Data Sharing in* ***Memory***
- ***Validate/Confirm*** *Results:* ***Profiling Cap Performance***

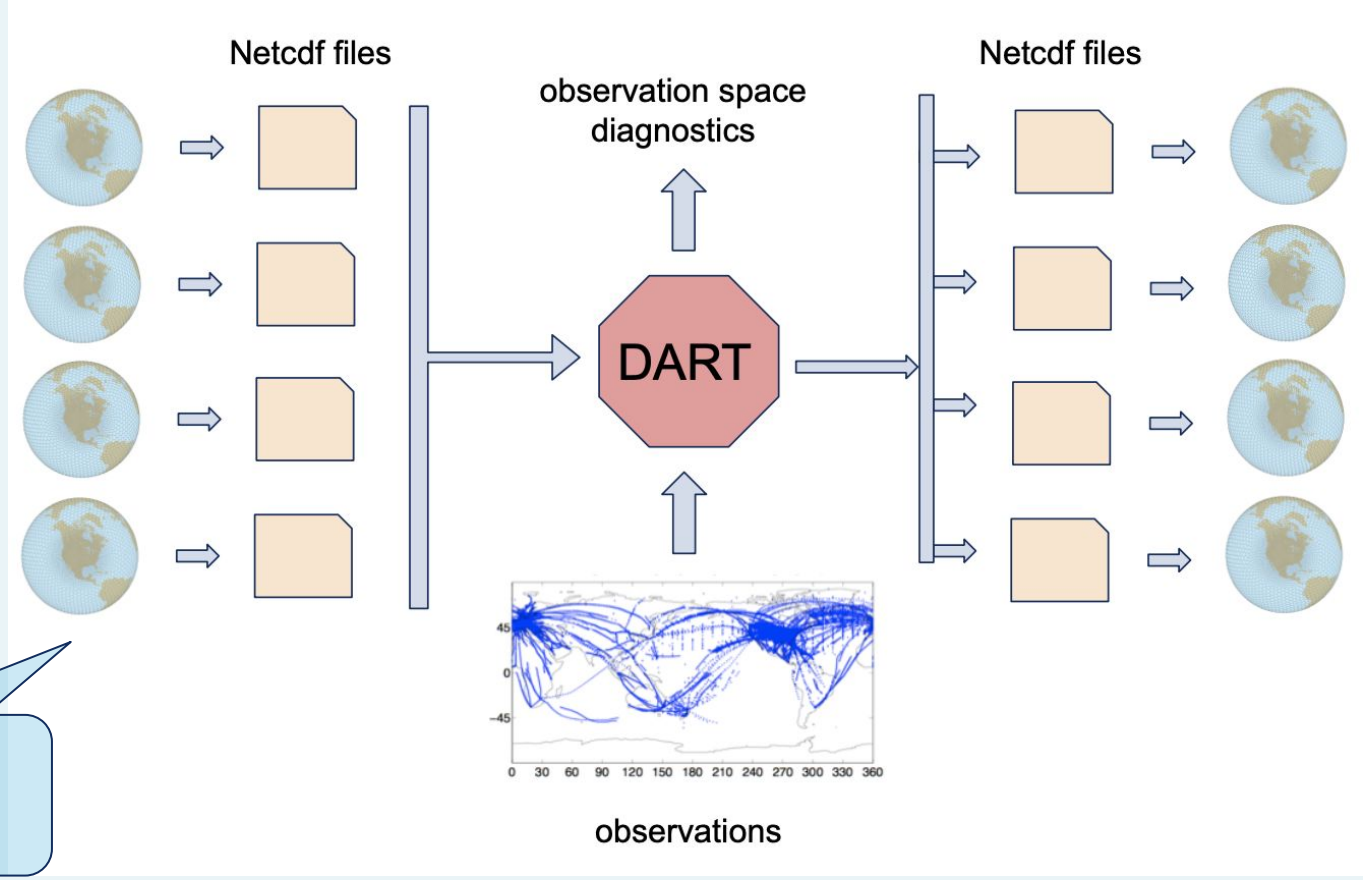
DART-CESM communication

DART needs:

- **Model states**
- Observations

to do data assimilation

How does DART talk to CESM (models)?

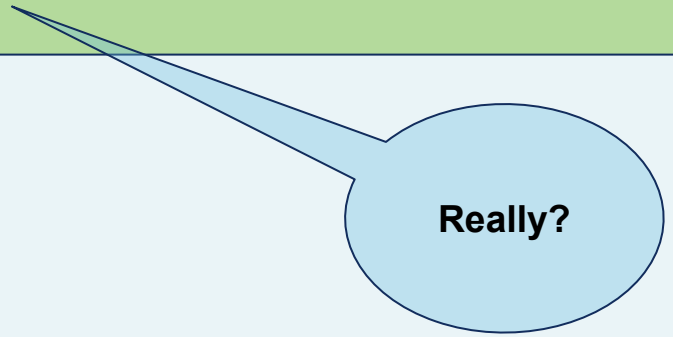


Project Objective

Objective: Build and test **DART's** ability to **access the model states <u>in memory</u>** using **NUOPC** (National Unified Operational Prediction Capability) thus avoiding the traditional **I/O bottlenecks** from file system data transfer.

Project Objective

Objective: Build and test **DART's** ability to **access the model states in memory** using **NUOPC** (National Unified Operational Prediction Capability) thus avoiding the traditional **I/O bottlenecks** from file system data transfer.

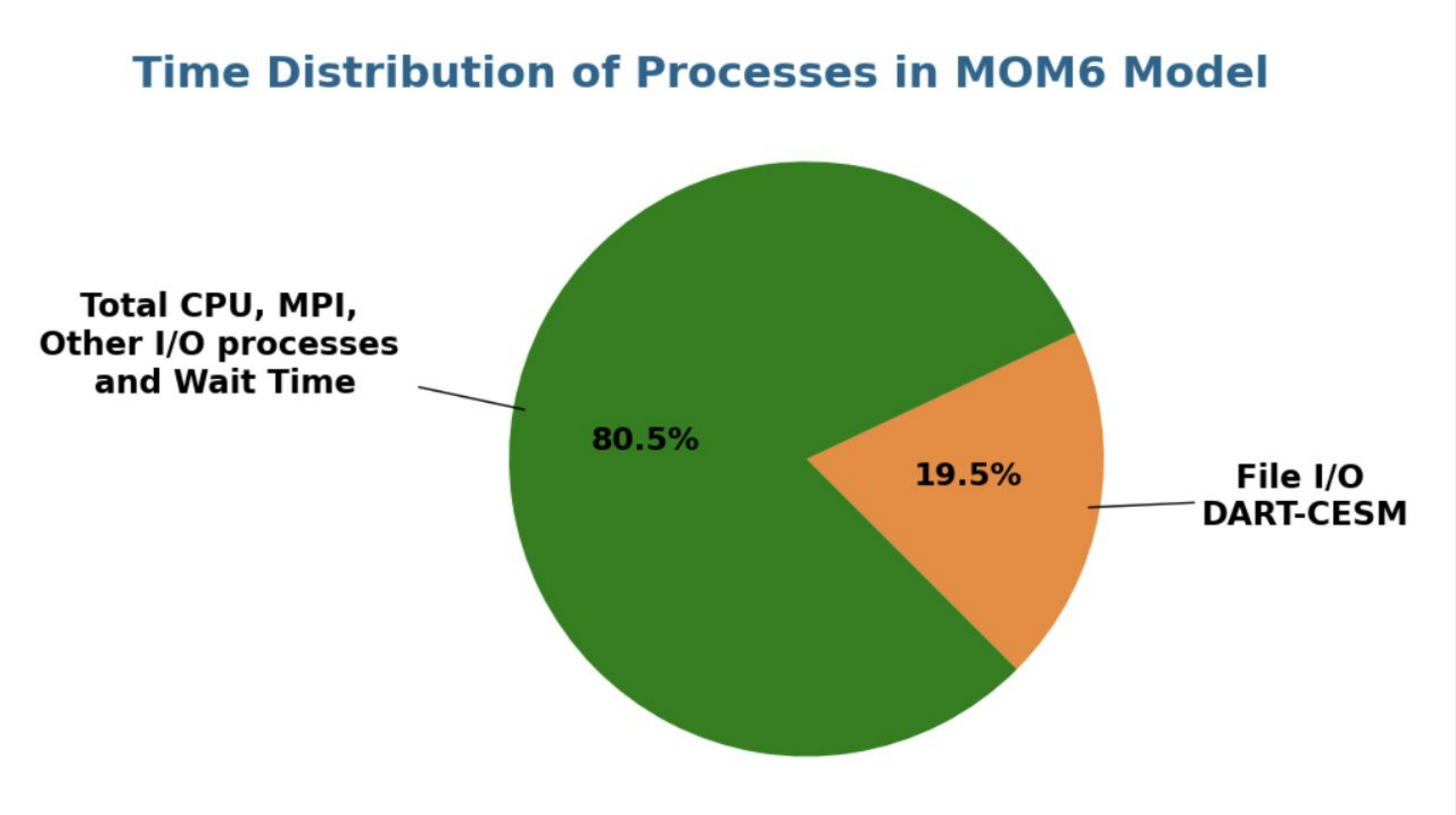


"peeking into the operating system"

Profiling  data exchange via the file system

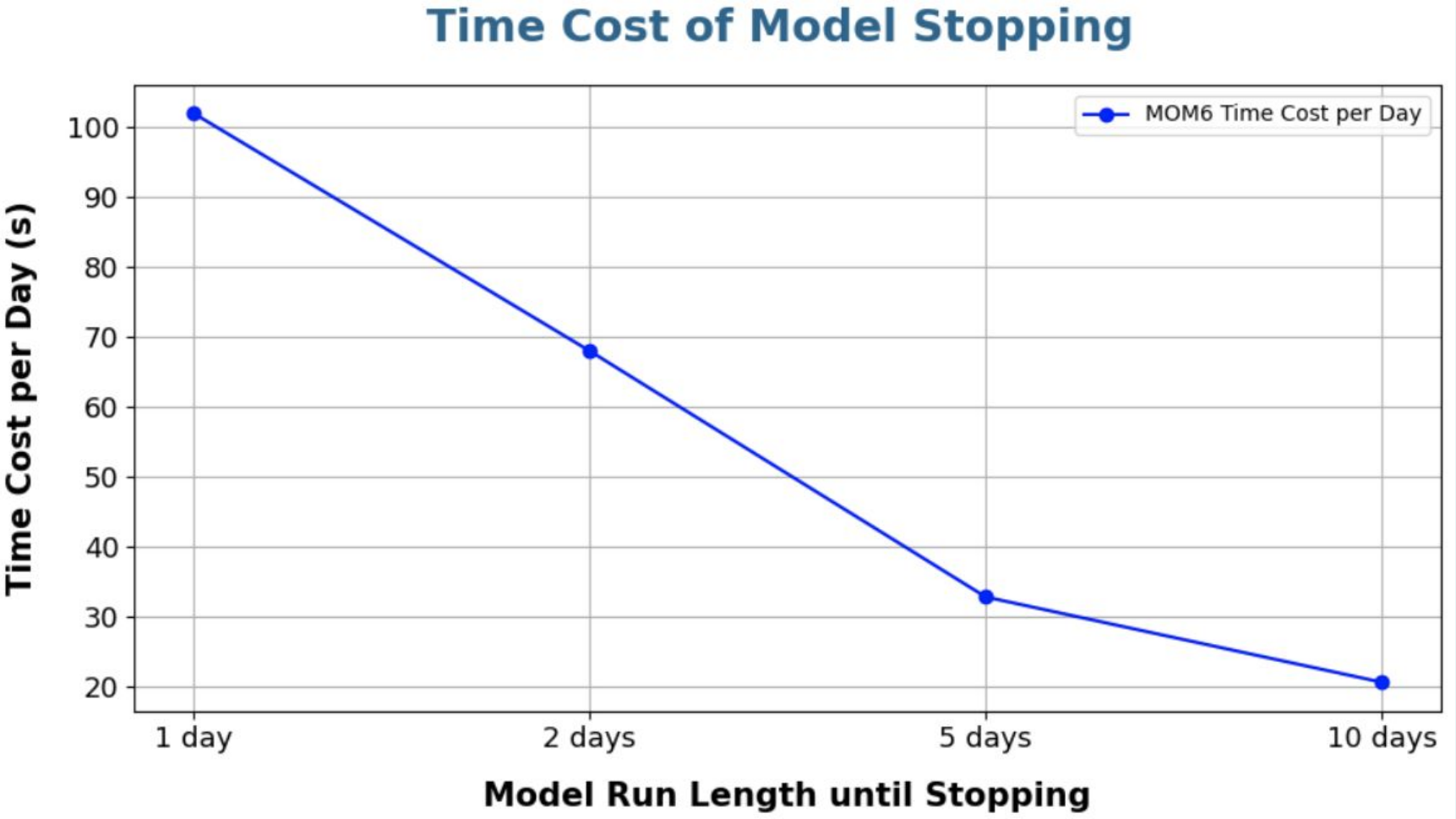
What happens when DART talks to CESM (models)?

Profiling Results



Benchmark profiling shows read/write operations consume significant runtime, even at relatively low-resolution (⅔ deg)

Profiling Results

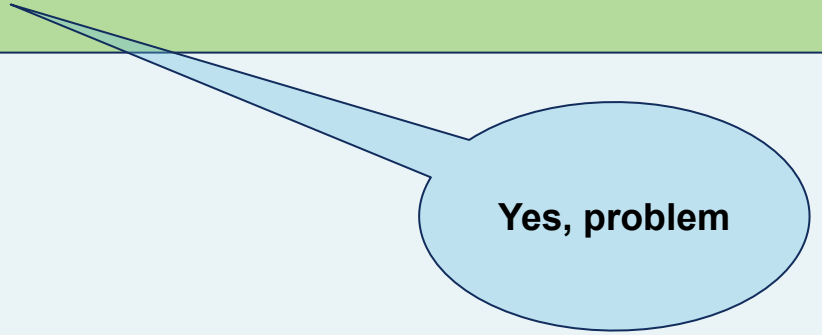


The more frequently we stop the model, the higher the fractional time cost.

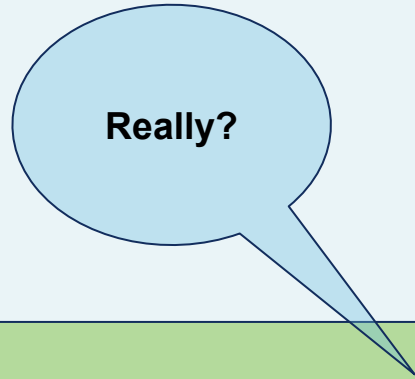
DA typically stops at least once per day.

Confirm I/O bottlenecks

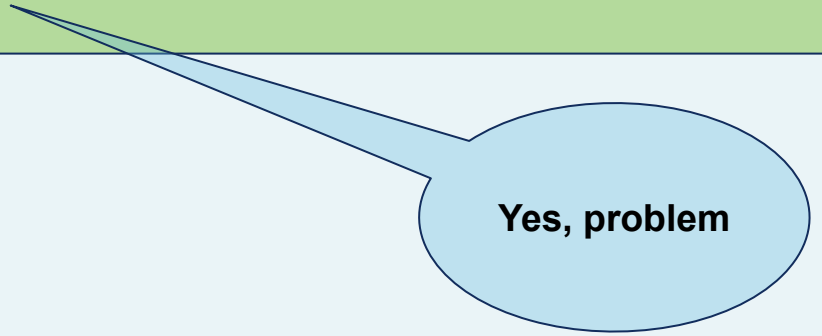
Objective: Build and test **DART's** ability to **access the model states in memory** using **NUOPC** (National Unified Operational Prediction Capability) thus avoiding the traditional **I/O bottlenecks** from file system data transfer.



Project Objective



Objective: Build and test **DART's** ability to **access the model states <u>in memory</u>** using **NUOPC** (National Unified Operational Prediction Capability) thus avoiding the traditional **I/O bottlenecks** from file system data transfer.



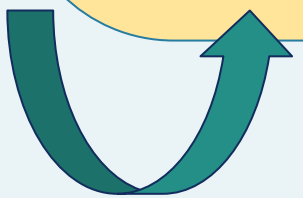
Outline

1. Introduction Background

- *Climate Modeling,* ***CESM***
- ***Data Assimilation***
- ***DART*** *Software*

2. Motivation Project Goals

- ***Profiling*** *results that shows* ***I/O bottlenecks***
- *Problem statement*
- *Why the* ***'Cap'*** *(****interface*** *for* ***CESM and DART****)?*



3. Methods Results

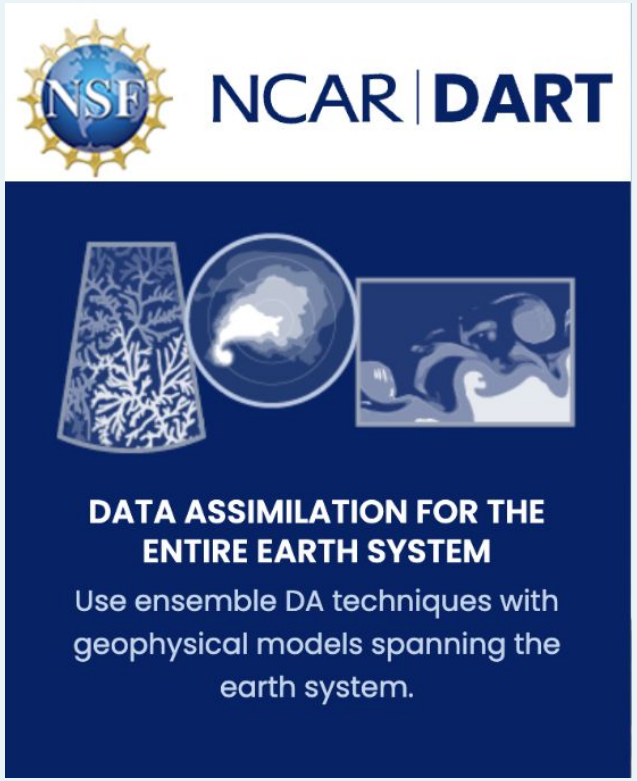
- ***Infrastructure*** *Challenges:*
  - *Derecho (HPC) vs. Docker (container)*
  - *ESMF, NUOPC, ESMX*
- ***ESMX*** *Framework decision*
- *The* ***Build Process*** *to* ***integrate*** *software, drivers and models for* ***Data Assimilation of Coupled Models***

4. Conclusions Future directions

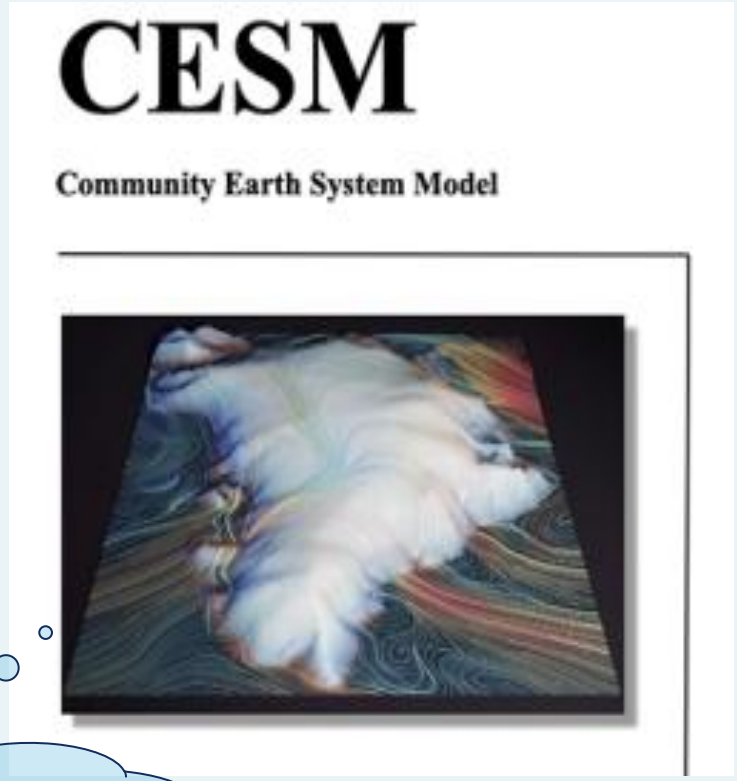
- *Advancing Climate Sciences:* ***Computational Frameworks and Software Infrastructure***
- *Direct Data Sharing in* ***Memory***
- ***Validate/Confirm*** *Results:* ***Profiling Cap Performance***

The Cap (interface)

DART



CESM



CAP

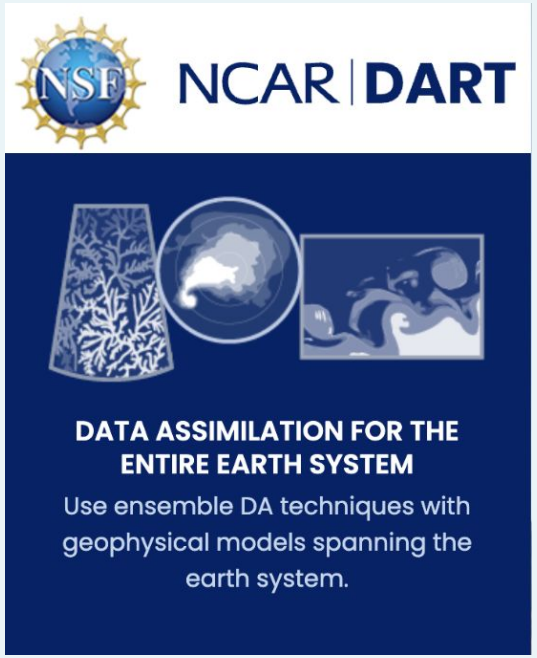
(a means for DART to access CESM model states in memory)

Coupled using **ESMF** (Earth System Modeling Framework)

Standardized using **NUOPC**

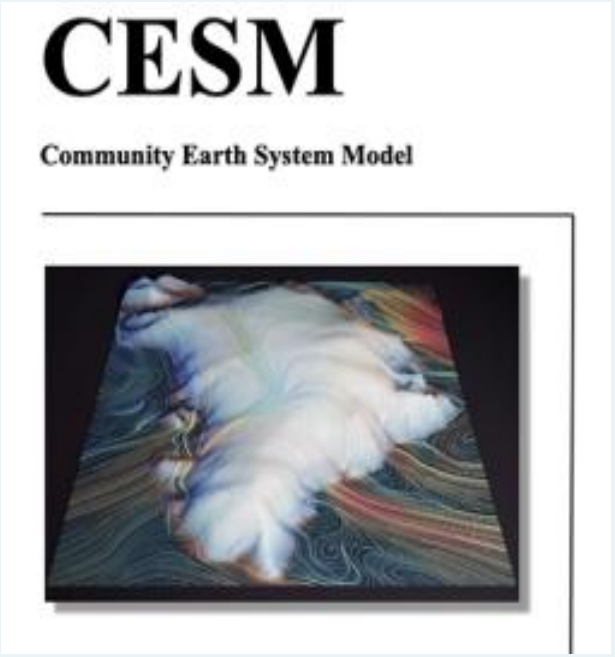
DART-NUOPC Cap

Objective: Build a wrapper (**cap**) as an **interface** to allow **DART** to **access the model states <u>in memory</u>** using **NUOPC** (National Unified Operational Prediction Capability)

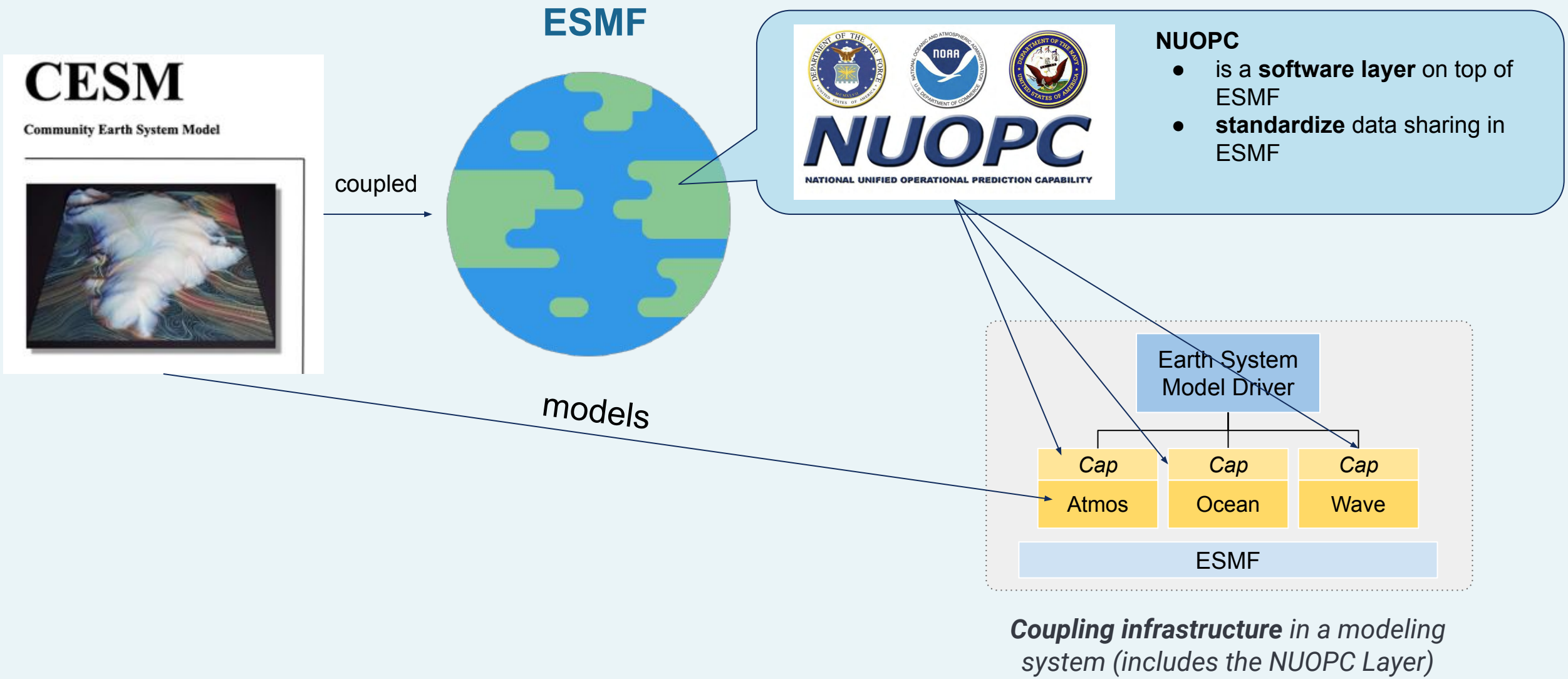


NUOPC Cap

~DART-NUOPC Cap
~ DART-CESM Cap



NUOPC Cap (translation layer)

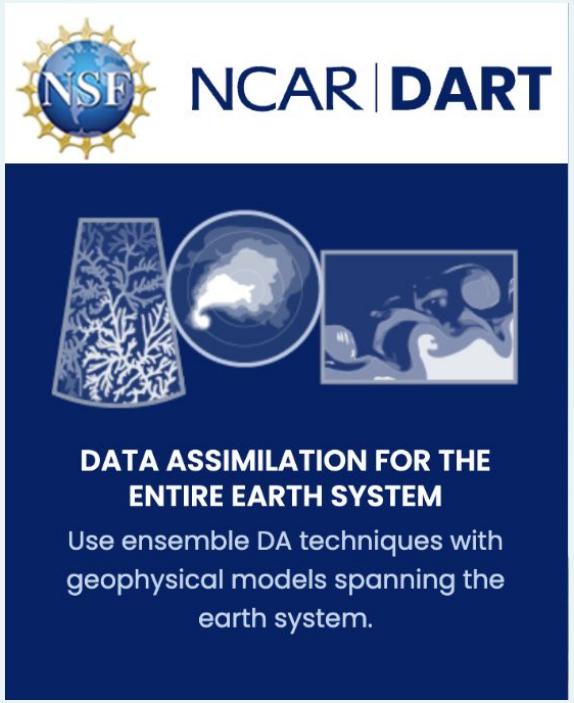


***Coupling infrastructure** in a modeling system (includes the NUOPC Layer)*

Why DART-NUOPC Cap?

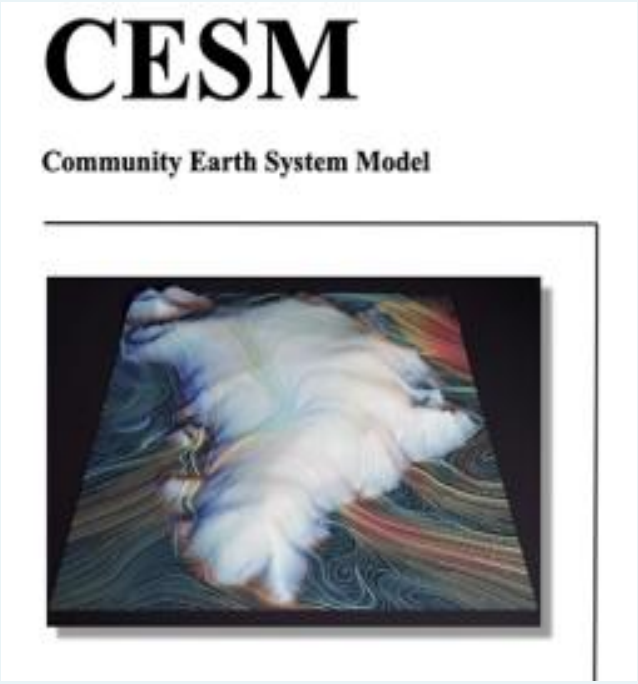
Q: Why don't we make changes internally to DART or CESM?
A: Maintainability, disruptions minimization, we just create a connection

Q: What does the cap do?
A: Standardize import, export data (handshake at initialization of fields)



NUOPC Cap

~DART-NUOPC Cap
~ DART-CESM Cap



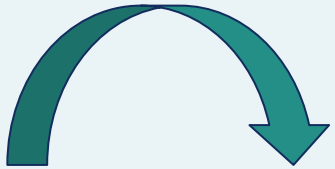
Outline

1. Introduction Background

- *Climate Modeling,* ***CESM***
- ***Data Assimilation***
- ***DART*** *Software*

2. Motivation Project Goals

- ***Profiling*** *results that shows* ***I/O bottlenecks***
- *Problem statement*
- *Why the* ***'Cap'*** *(****interface*** *for* ***CESM and DART****)?*



3. Methods Results

- ***Infrastructure*** *Challenges:*
  - *Derecho (HPC) vs. Docker (container)*
  - *ESMF, NUOPC, ESMX*
- ***ESMX*** *Framework decision*
- *The* ***Build Process*** *to* ***integrate*** *software, drivers and models for* ***Data Assimilation of Coupled Models***

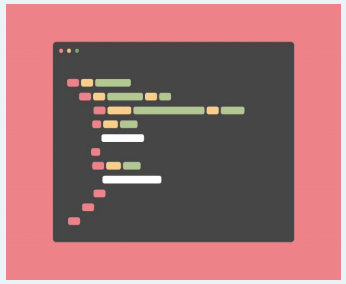
4. Conclusions Future directions

- *Advancing Climate Sciences:* ***Computational Frameworks and Software Infrastructure***
- *Direct Data Sharing in* ***Memory***
- ***Validate/Confirm*** *Results:* ***Profiling Cap Performance***

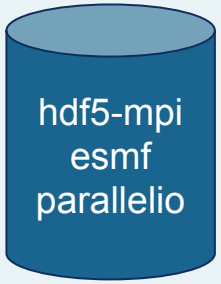
Software Infrastructure

What is Software Infrastructure?

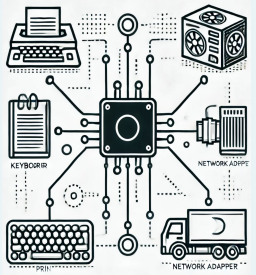
Code



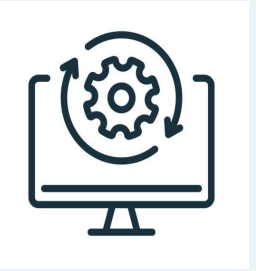
Libraries



Drivers



Operating System



Hardware

Tasks requested by code

Why Software Infrastructure?

- Programmers always make infrastructure choices
- Different components/programs have different views
- Communication among all components requires **coordination** and **optimization**
- Different softwares and systems are not built to work together, how do we facilitate **integration of distinct softwares**?

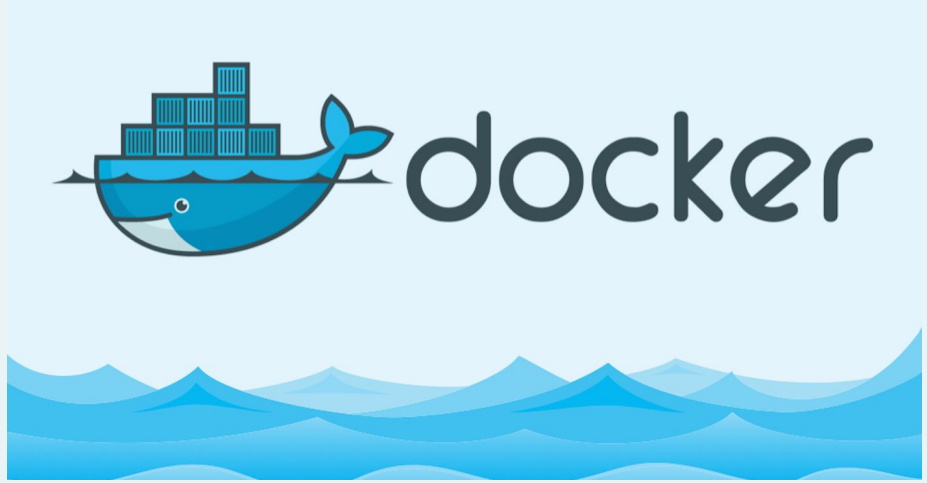
Choosing Tools and Frameworks

1. Environment Selection: Docker vs. Derecho

- **Derecho:** NCAR's new HPE Cray EX cluster (**supercomputer** comprised of interconnected nodes)

➢ **Centralized Tool for Integration**

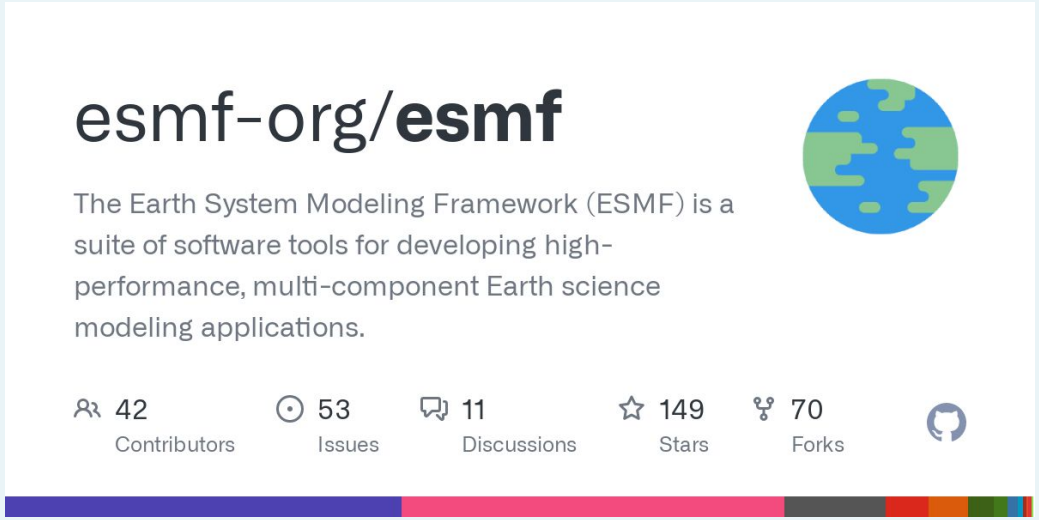
- **Docker:** container to deploy, manage, and run applications in **isolated environments**



2. Framework: ESMF vs. ESMX

ESMF Framework

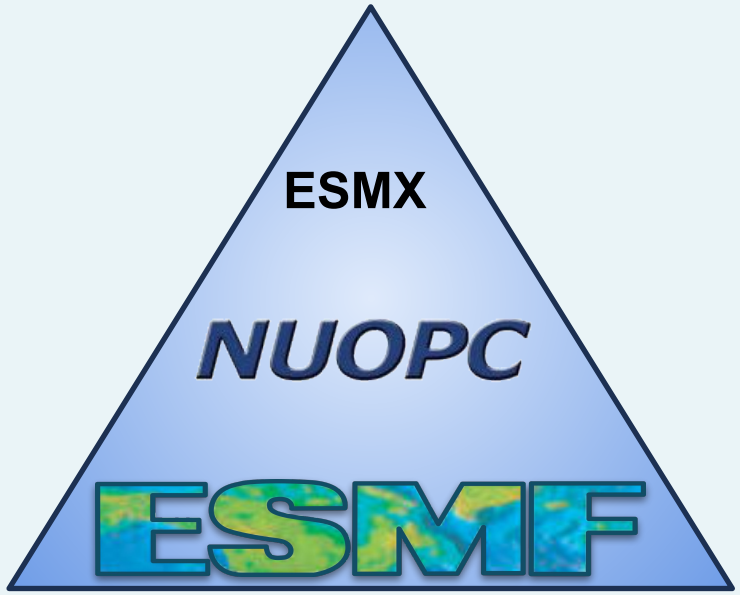
- **Pros**: can facilitate building DART cap with all CESM (end goal)
- **Cons**:
  - ➢ Sophisticated infrastructure
  - ➢ Domain knowledge
  - ➢ Time constraints
    external lab (ESMF) communication



Reducing the problem: ESMF vs. ESMX

ESMX

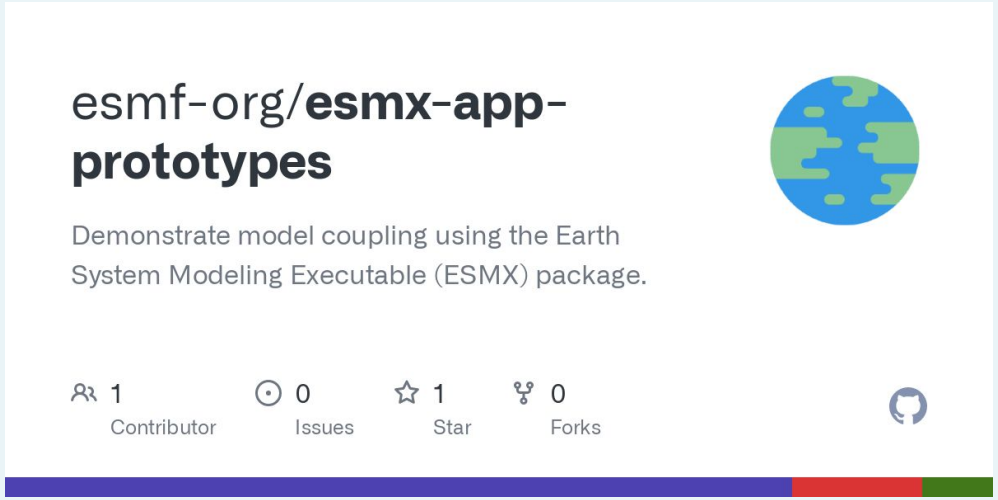
- **ESMX**: Earth System Model eXecutable **layer** built on top of ESMF and NUOPC APIs
- **Purpose**: **Simplifies** building, running, and testing NUOPC-based systems
  - ➢ **Orchestration** handled by ESMX
    ⇨ No need to write drivers!
  - ➢ **Streamlines** compiling and linking components (in YAML language)



Software Stack

Why ESMX?

ESMX is a framework for testing and developing **cap on its own**, before integrating a full system



3. Model Component: CDEPS

CDEPS
(Community Data-Model Evaluation and Prediction System)

- Acts as a **model component** without the need for a fully coupled system model

Aspect	CDEPS	Full Models
Complexity	Simplified setup	High complexity
Focus	Specific components	Entire system
Testing	Targeted component testing	Whole system testing

Criteria for the Build Process (towards Optimization)

Criteria for the Build Process

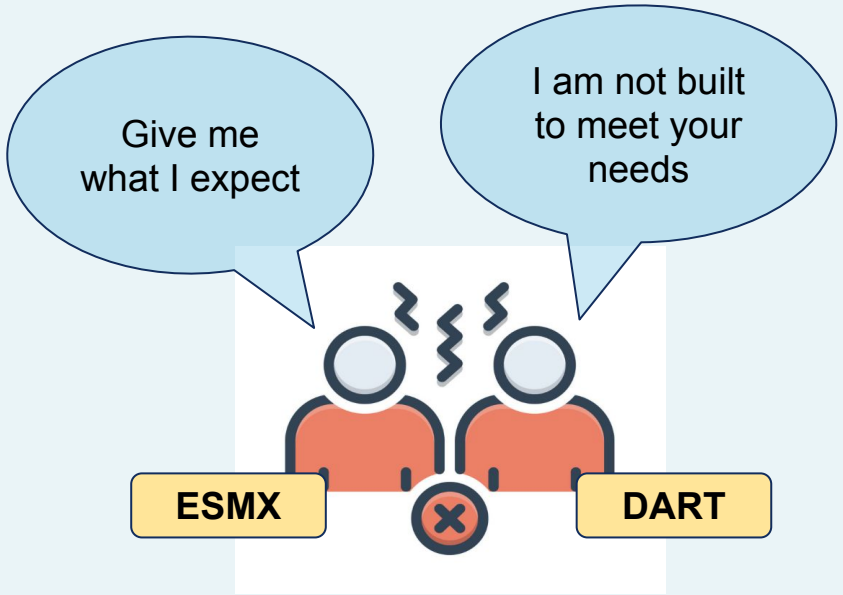
1. Streamlining: **simplifying/removing** unnecessary elements/steps (e.g. ESMX, CDEPS)

2. Minimizing Disruptions (maintainability): Make **changes**, **integrate** systems **without disrupting** operations or impacting existing features (e.g. the cap approach, ESMX)

Criteria for the Build Process

3. System Compatibility:

- Systems/software aren't always built to work together
- Are multiple software expecting the same object form?
  Are they operating on a shared infrastructure?
  Security, access control measures, data formats, etc.
- e.g. ESMX and DART



4. Constraints Consideration

Outline

1. Introduction Background

- *Climate Modeling,* ***CESM***
- ***Data Assimilation***
- ***DART*** *Software*

2. Motivation Project Goals

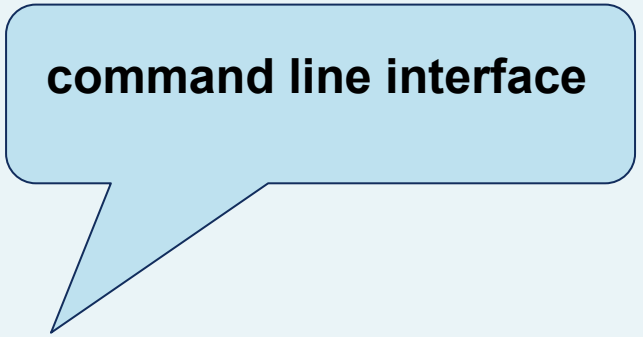
- ***Profiling*** *results that shows* ***I/O bottlenecks***
- *Problem statement*
- *Why the* ***'Cap'*** *(****interface*** *for* ***CESM and DART****)?*

3. Methods Results

- ***Infrastructure*** *Challenges:*
  - *Derecho (HPC) vs. Docker (container)*
  - *ESMF, NUOPC, ESMX*
- ***ESMX*** *Framework decision*
- *The* ***Build Process*** *to* ***integrate*** *software, drivers and models for* ***Data Assimilation of Coupled Models***

4. Conclusions Future directions

- *Advancing Climate Sciences:* ***Computational Frameworks and Software Infrastructure***
- *Direct Data Sharing in* ***Memory***
- ***Validate/Confirm*** *Results:* ***Profiling Cap Performance***



The Build Process

(integrate softwares, drivers and models for DART-CESM communication aka the cap)

1. Build Dependencies

- What **compilers** work with what **machines?**

  What **libraries** with **High Performance Computing** (HPC)?

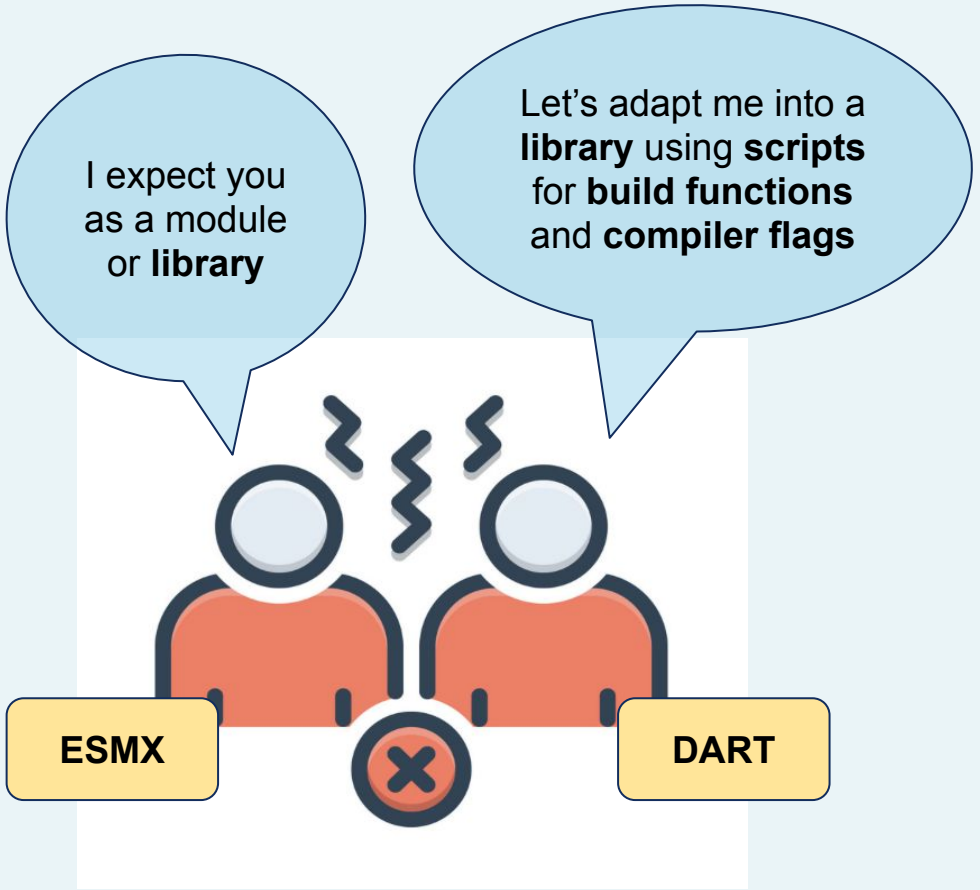
  Support parallel processing, scaling, synchronization

- How: **build templates**

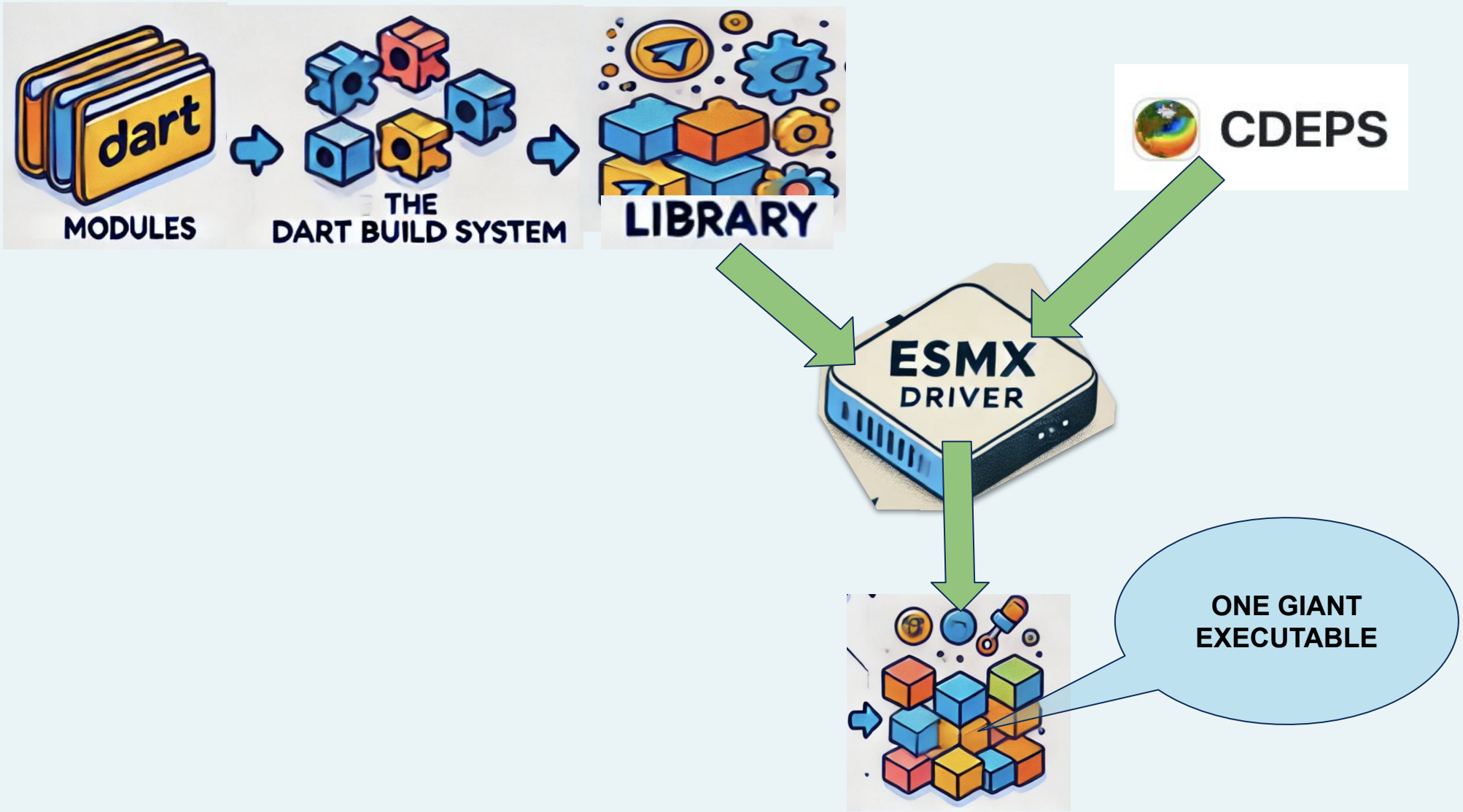
2. Transform and Unify Objects

- **Consistent Data Forms:** Match the output of one software to the expected input of another.
  - ➢ ESMX requires components to be compiled as **libraries** for a unified executable.
  - ➢ DART is outputting **executables**

➡ **DART Integration**: modify and incorporate DART as a library so ESMX can digest it



DART Executables => DART library



3. Build Configurations

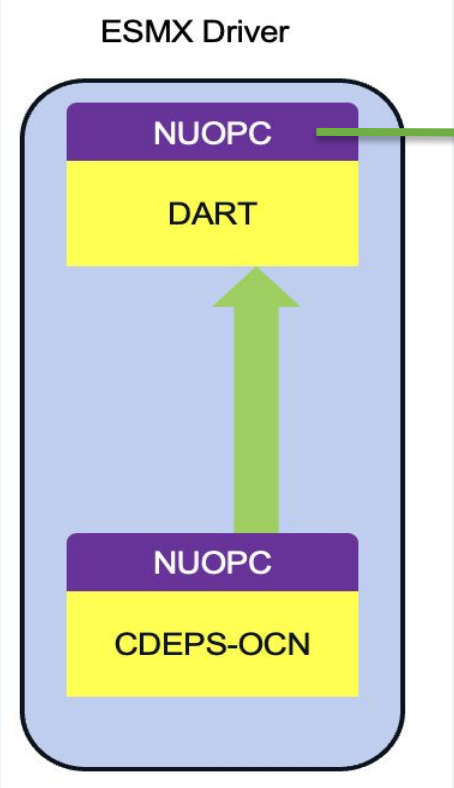
- **Locating Software Components**
  - ➢ *esmxBuild.yaml* file

DART as a component

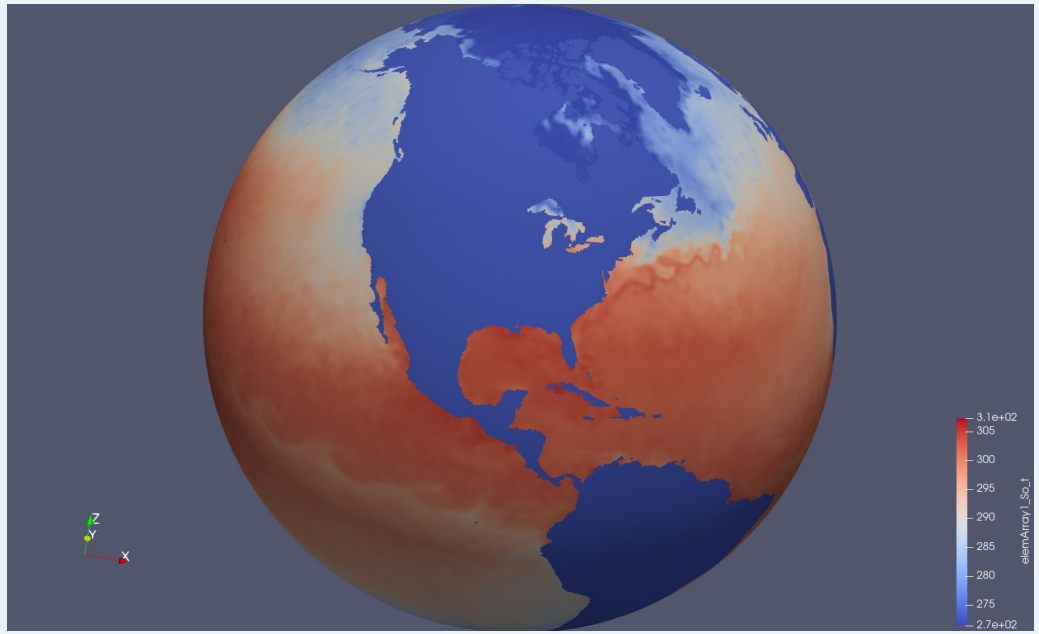
```
dart:
  build_type: script
  build_script: quickbuild.sh
  source_dir: /glade/u/home/sumansh/Altrun/DART/models/cdeps/work
  install_prefix: /glade/u/home/sumansh/Altrun/DART/models/cdeps/work
  fort_module: dart_comp_nuopc.mod
```

CAP

Final Results: First in-memory DART ⟷ CESM



Subroutine StateWriteVTK( )



ParaView plot confirms DART receives the 2D field

Outline

1. Introduction Background

- *Climate Modeling,* ***CESM***
- ***Data Assimilation***
- ***DART*** *Software*

2. Motivation Project Goals

- ***Profiling*** *results that shows* ***I/O bottlenecks***
- *Problem statement*
- *Why the* ***'Cap'*** *(****interface*** *for* ***CESM and DART****)?*

3. Methods Results

- ***Infrastructure*** *Challenges:*
  - *Derecho (HPC) vs. Docker (container)*
  - *ESMF, NUOPC, ESMX*
- ***ESMX*** *Framework decision*
- *The* ***Build Process*** *to* ***integrate*** *software, drivers and models for* ***Data Assimilation of Coupled Models***

4. Conclusions Future directions

- *Advancing Climate Sciences:* ***Computational Frameworks and Software Infrastructure***
- *Direct Data Sharing in* ***Memory***
- ***Validate/Confirm*** *Results:* ***Profiling Cap Performance***

Summary and Conclusions

- Developing robust **software infrastructure** minimizes disruptions and enables new feature development and testing. **DART-X** supports the DART-NUOPC Cap, the first in-memory data transfer prototype between DART and CESM.

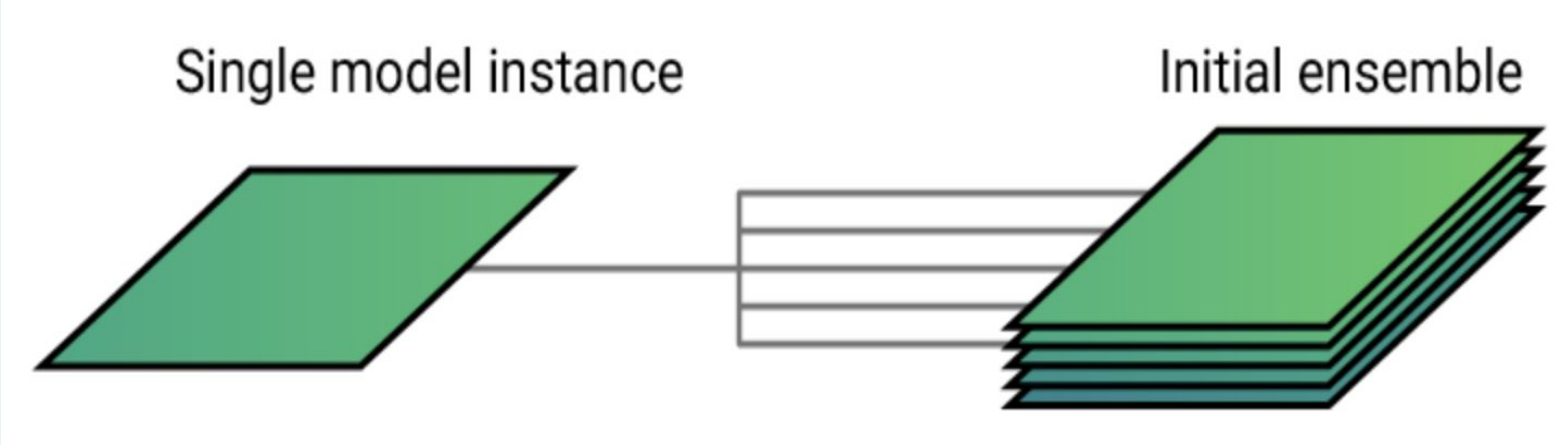
- The **first DART-NUOPC cap prototype** paves the way to explore **direct memory sharing,** potentially reducing disk I/O bottlenecks. This enables DART to **act as a model component**, speeding up data assimilation and reducing computational costs.

Full DART-ESMF integration

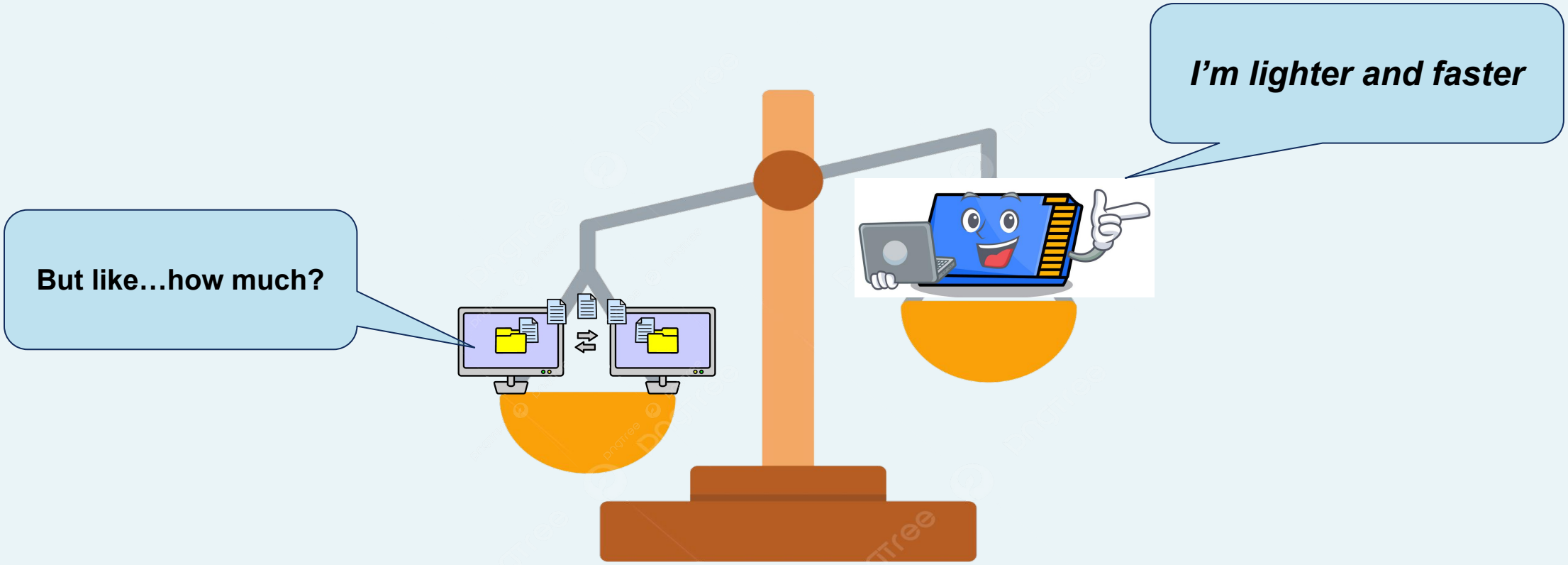
- **Advance** from a successful ESMX prototype, **DART-X**, to a **full DART-CESM cap** using EMSF driver.

Ensemble Field Transfers

- Continue testing DART as a model component for **ensemble models** to transfer **ensemble fields.**

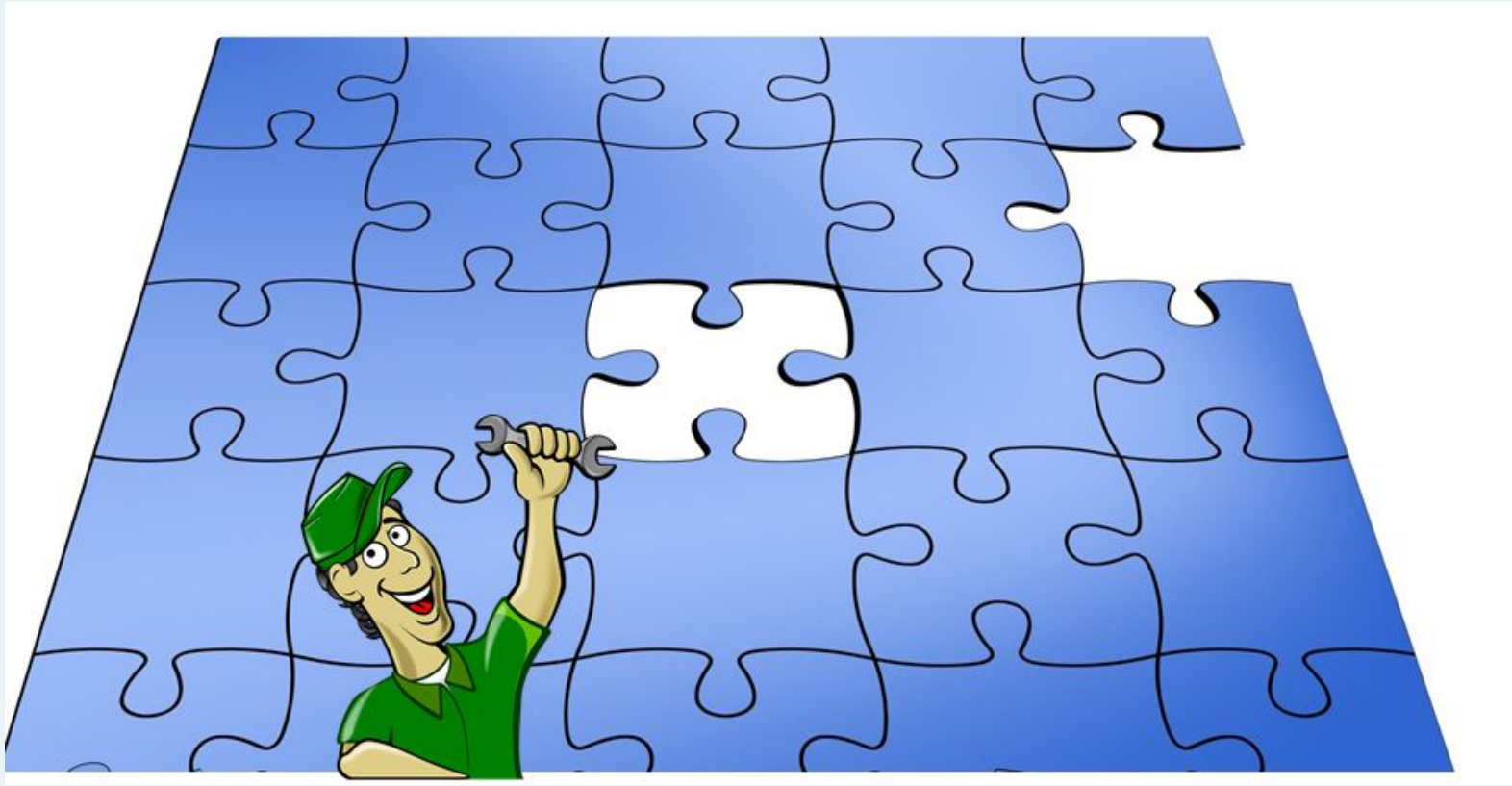


Profiling: Disk vs. Memory



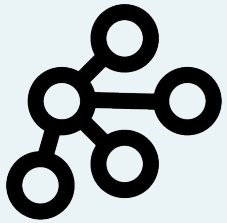
Handling Missing Fields (data) In Memory

The cap currently exchanges fields; future work will address missing data in DART-CESM communication.



Future Directions

Generalize in-memory data passing to other data assimilation systems, data-related applications, and potentially machine learning frameworks



Acknowledgement and Gratitude

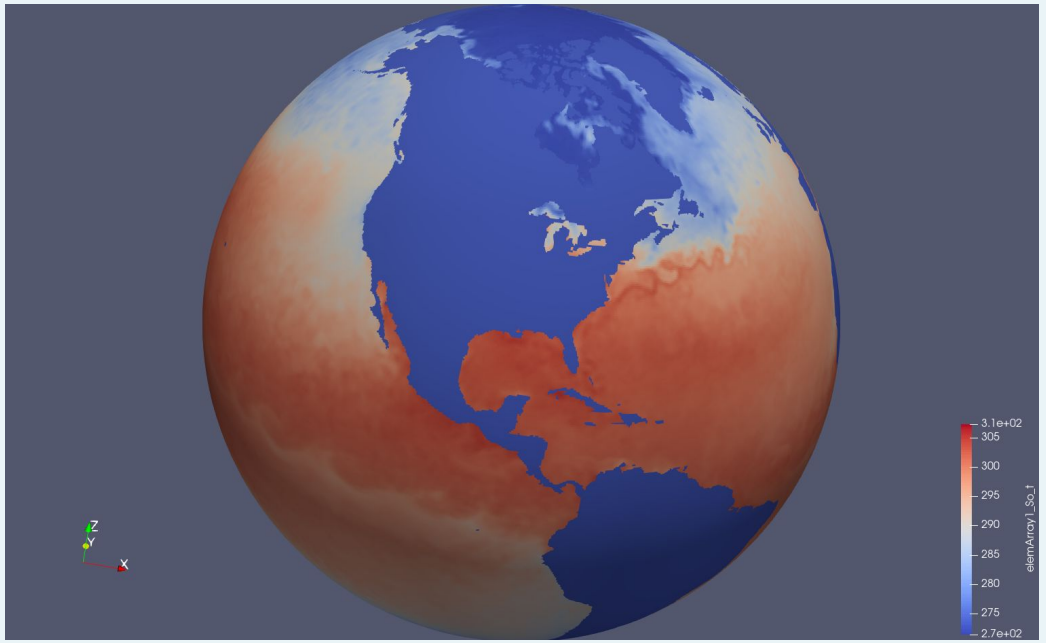
- Project partner: **Suman Shekhar** for his contribution on the first in-memory data transfer prototype for DART-CESM. I admire his dedication, tremendous passion for climate sciences, and collaborative spirit.
- Mentors: **Helen Kershaw, Dan Amrhein, Ufuk Turuncoglu**
  I am grateful for their expertise, vision, and their generous support.
- **DAReS: Jeffrey Anderson, Moha Gharamti, Kevin Raeder, Helen Kershaw, Marlee Smith, Ben Johnson, Ann Norcio** for their technical support on DART.
- **ESMF: Dan Rosen, Ann Tsay, Jim Edwards, Bill Sacks, Ufuk Turuncoglu** for their technical support on ESMF.
- **SIParCS** Admin
  A special thanks to SIParCS Program Director, **Virginia Do**, for her care, vision and support.
- **CISL, NCAR, UCAR NSF** for funding and facilities.
- **Fellow SIParCS technical interns** for their collaboration, enjoyable activities, and the opportunity to connect and share knowledge, and CODE intern **Eva Sosoo** for her community engagement.
- **NESSI** cohort for shared activities with our SIParCS cohort, **Jerry Cyccone**, **Benjamin Fellman, Jessica Wang** for their organizing.

Q&A

Optimizing Ensemble Data Assimilation for Coupled Earth System Models

DART-X: Software Infrastructure for Prototyping in-memory Data Transfer between Ensemble Data Assimilation and Coupled Earth Systems Models

DART ⟷ **CESM**



Email: pham36a@mtholyoke.edu **LinkedIn:**