

# Reliability Reproducibility Replicability

## in Climate Modelling

Aidan Heerdegen

## Credits:

Harshula Jayasuriya (ACCESS-NRI)

Tommy Gatti (ACCESS-NRI)

Varvara Efremova (ACCESS-NRI)

Jeroen van den Muyzenberg (ACCESS-NRI)

Johanna Basevi (ACCESS-NRI)

Utkarsh Gupta (ACCESS-NRI)

Micael Oliveira (COSIMA)

Scott Wales (Bureau of Meteorology)

Angus Gibson (ANU)



## Introduction

**Reliability:** it just works

**Reproducibility:** answers don't change

**Replicability:** verify robust results

All three improve trust

Trust in infrastructure and trust in the model results

# ACCESS-NRI

## Australian Earth System Simulator

Modelling weather and climate to support research and decision-making

### WHAT WE DO



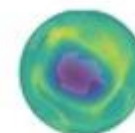
**Create** and **curate** open-source software infrastructure in collaboration with Australian research communities



**Build** the capability of Australian in climate science, observations and high performance computational modelling research



**Support** software, tools and documentation for the ACCESS community



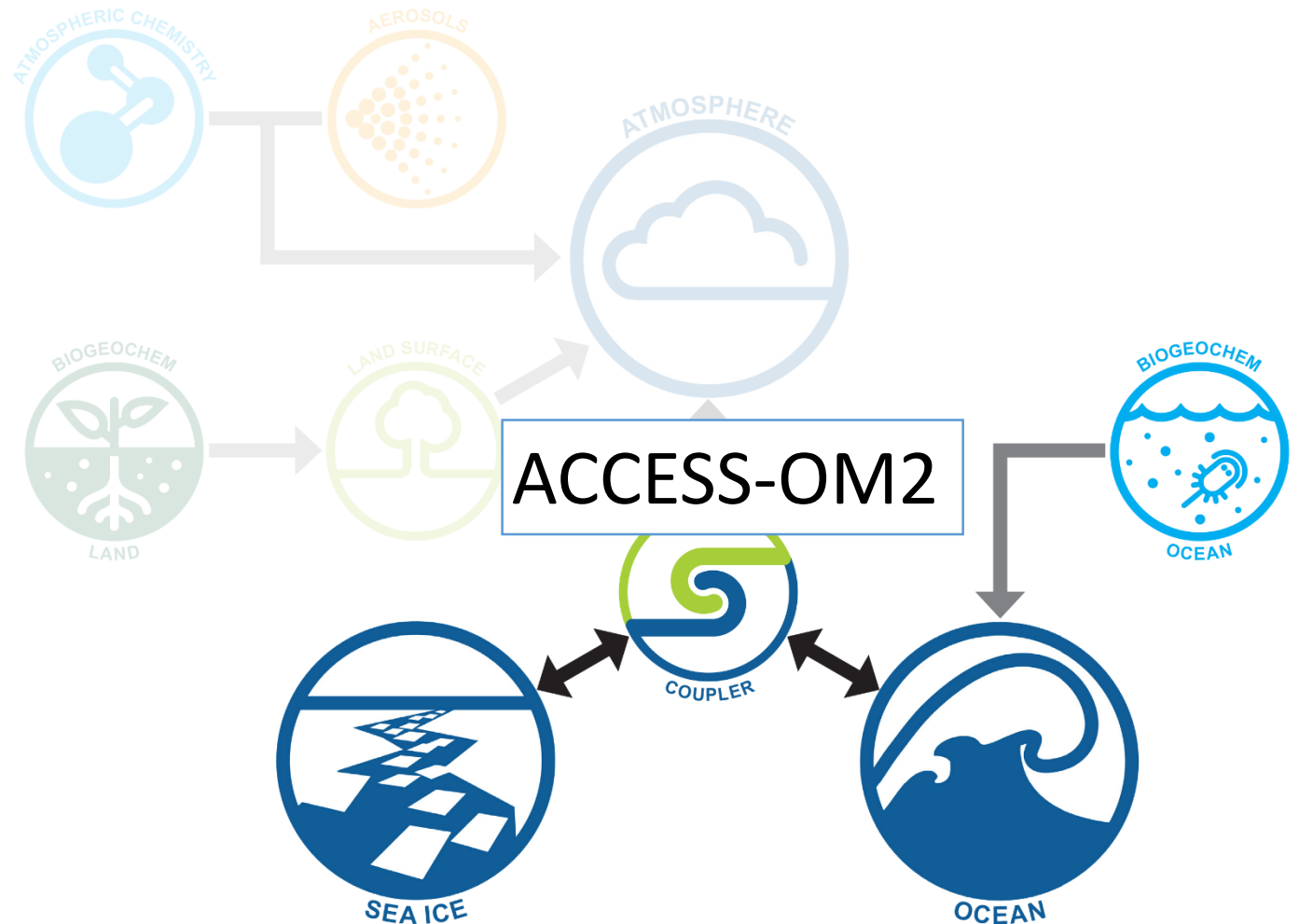
**Develop** & **configure** transparent, quality-assured climate prediction systems and **enhance** current Australian climate modelling systems

# What is the ACCESS model?

ACCESS is a computer Earth system modelling framework, made up of a number of different components.

Our models simulate different future weather and climate scenarios by linking separate model components that represent different parts of the Earth System.

These model components can be run in isolation, or they can be joined together to create coupled model configurations.





# Who is the ACCESS Community?

The ACCESS Community is researchers from the Australian Climate Research Community: Universities, Bureau of Meteorology, CSIRO and other Government Agencies.

NCI is the main platform for collaboration and where resource intensive simulations are run and the resulting data stored and analysed.

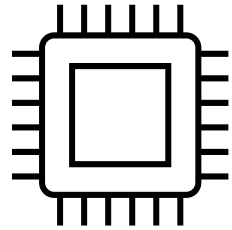
# Resource Requirements

ACCESS-OM2-01

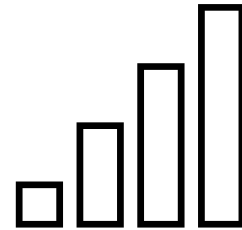


100  
model  
years

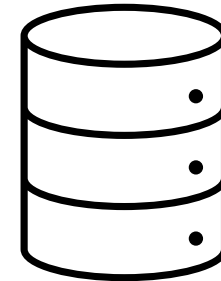
=



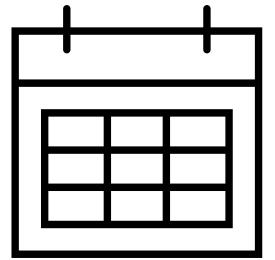
10K+  
cores




25 MSU



100TB



3 Months



*"Climate models are mathematical equations that describe how the important parts of the earth's atmosphere, ocean and biology act and interact to produce the world we live in. They are tools that can be used to ask questions about why the climate has changed, and how it may change in the future based on decisions we make now.*

***They have a track record of being correct"***

- Scott St. George, Head of Weather and Climate Research, WTW



## Goal

Scientists can trust that any variation in a climate model is due to factors under their control, rather than changes in software dependencies, or the tools used to build the model.



**Reliability**  
**Reproducibility**  
**Replicability**



# Reliability: it just works

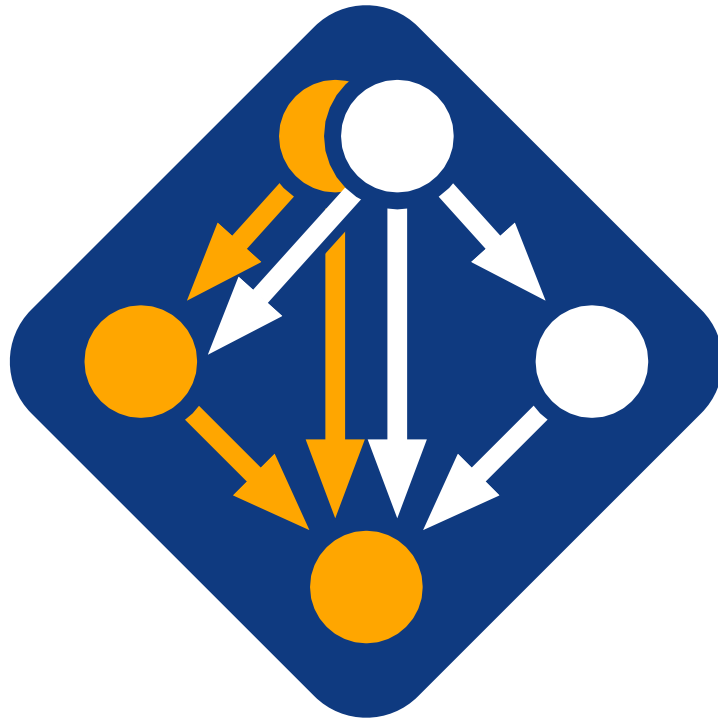
- Defined support timeline, and nature of support
- Released models maintained in good order (performance, compatibility, bug fixes)
- Thorough documentation
- Full provenance of code, build, configuration and inputs
- Extensive and discoverable model evaluation
- Model versions have been tested, evaluated and deployed correctly (versions correspond to those tested)



*"Do you have the **good binary**?"*

- One scientist to another

# Better Tools




Developed at LLNL  
**7000** packages  
**1100** contributors  
Popular and widespread

## Spack

<https://spack.io>





spec: spack has a highly expressive syntax spec that allows for complete customisation of builds including compiler and dependency versions.

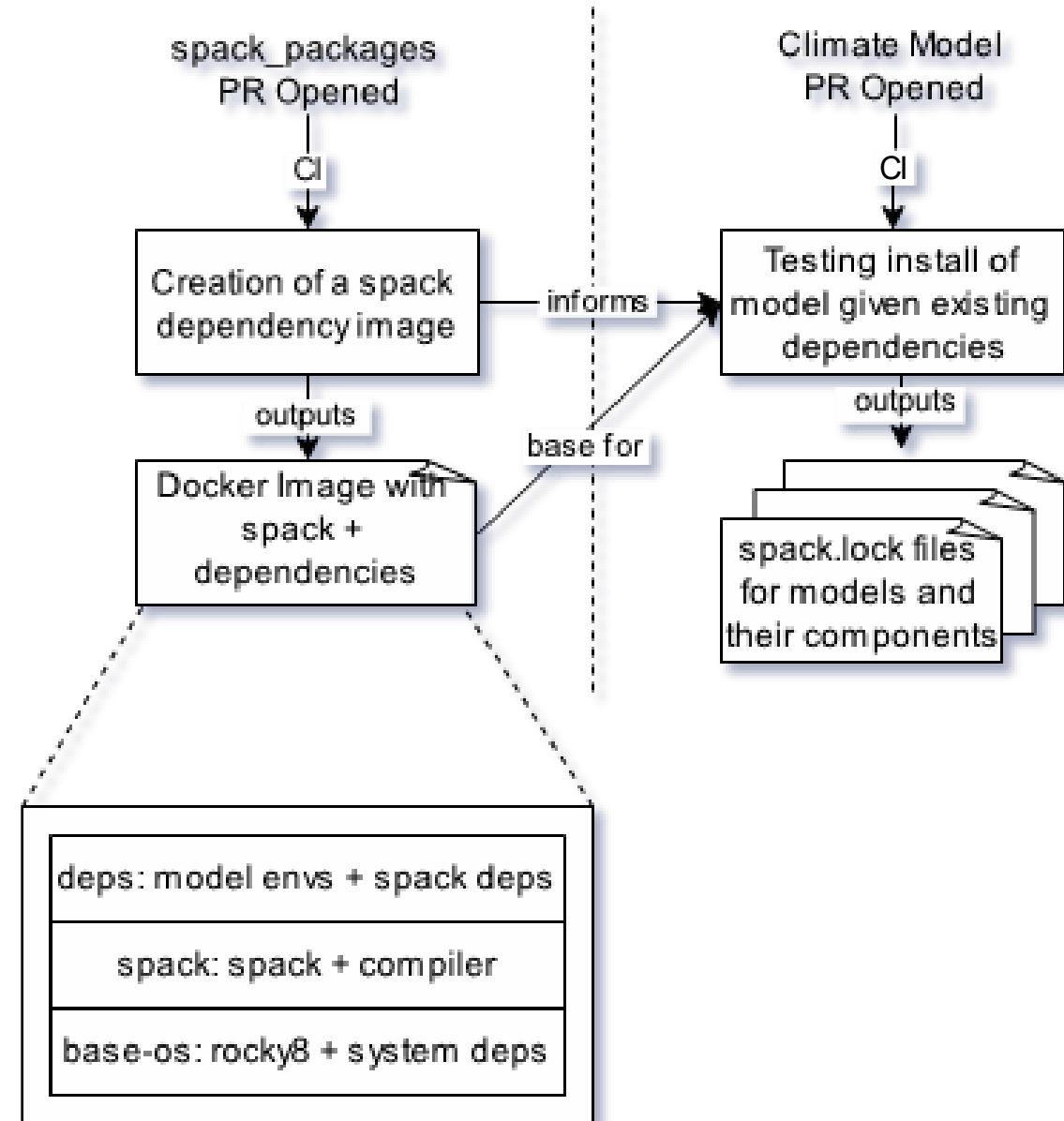
```
spack install access-om2 %intel@2021.1.2 \  
^nci-openmpi@4.0.2 ^netcdf-c@4.7.4 ^netcdf-fortran@4.5.2
```

Same spec can be used to build a model on an HPC, a GitHub runner, and inside a container or on your laptop: can create containers for local development work

```
docker build -f Dockerfile.base-spack -t spack-dev:latest \  
\ --target=dev .  
docker run -it --rm spack-dev:latest
```

# CI: GitHub, Workflows, Containers and Spack

- **Issue:** three hours to build dependencies
- **Fix:** CI/CD:
  - Create dependency docker images (spack + common ACCESS model dependencies)
  - Multiple caches – GitHub Actions job caching, docker image caching, spack build caching
  - Install models and their components on top of dependency images
- **Result:** 1-2 mins to install CI compile testing dependencies
- Model tests output `spack.lock` files that can regenerate exact environment





# Reproducibility: answers don't change

Same model, build, configuration and inputs running on the same hardware should produce exactly the same answers, bit for bit

This is known as **bitwise reproducible**

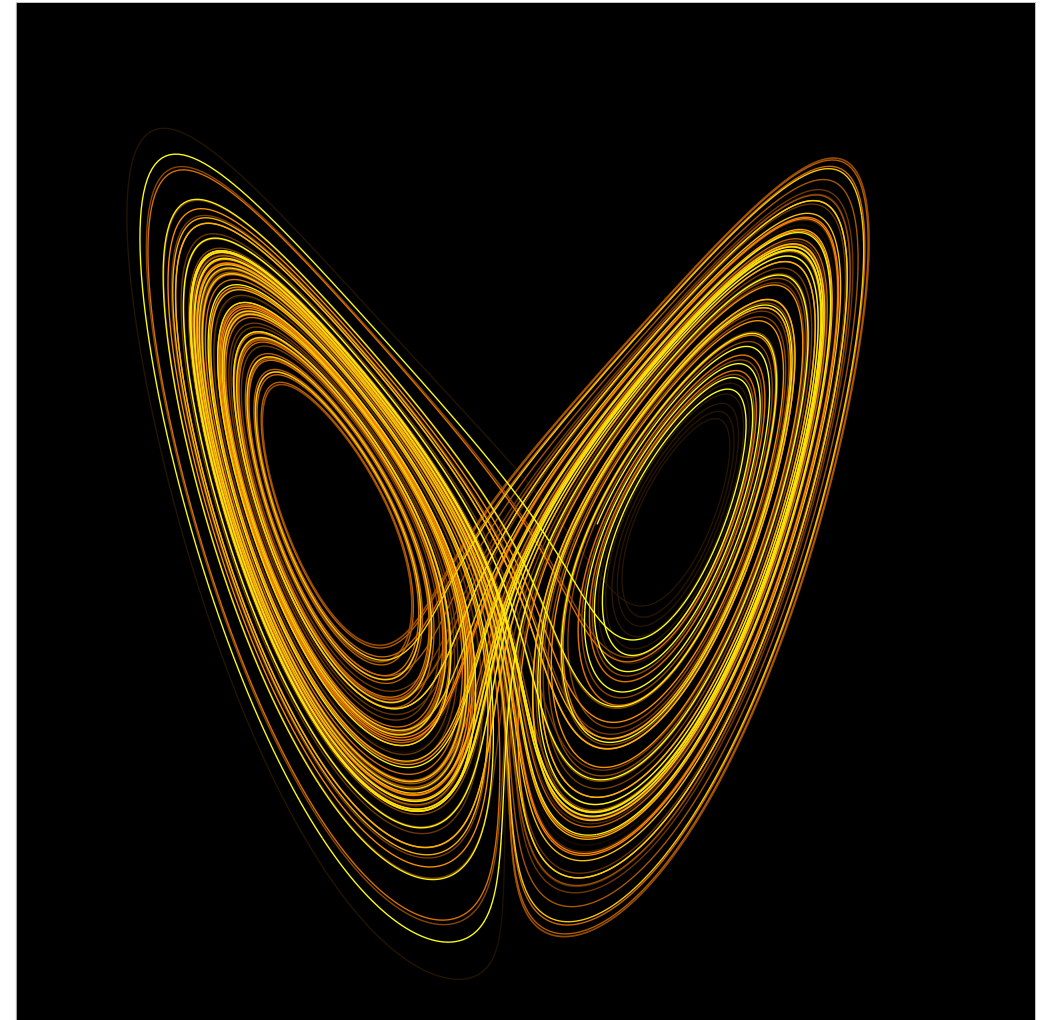
# Climate models are Chaotic (good)

Chaos theory discovered in 1961 by Edward Lorenz while working on very early atmosphere model

Coined “Butterfly Effect”

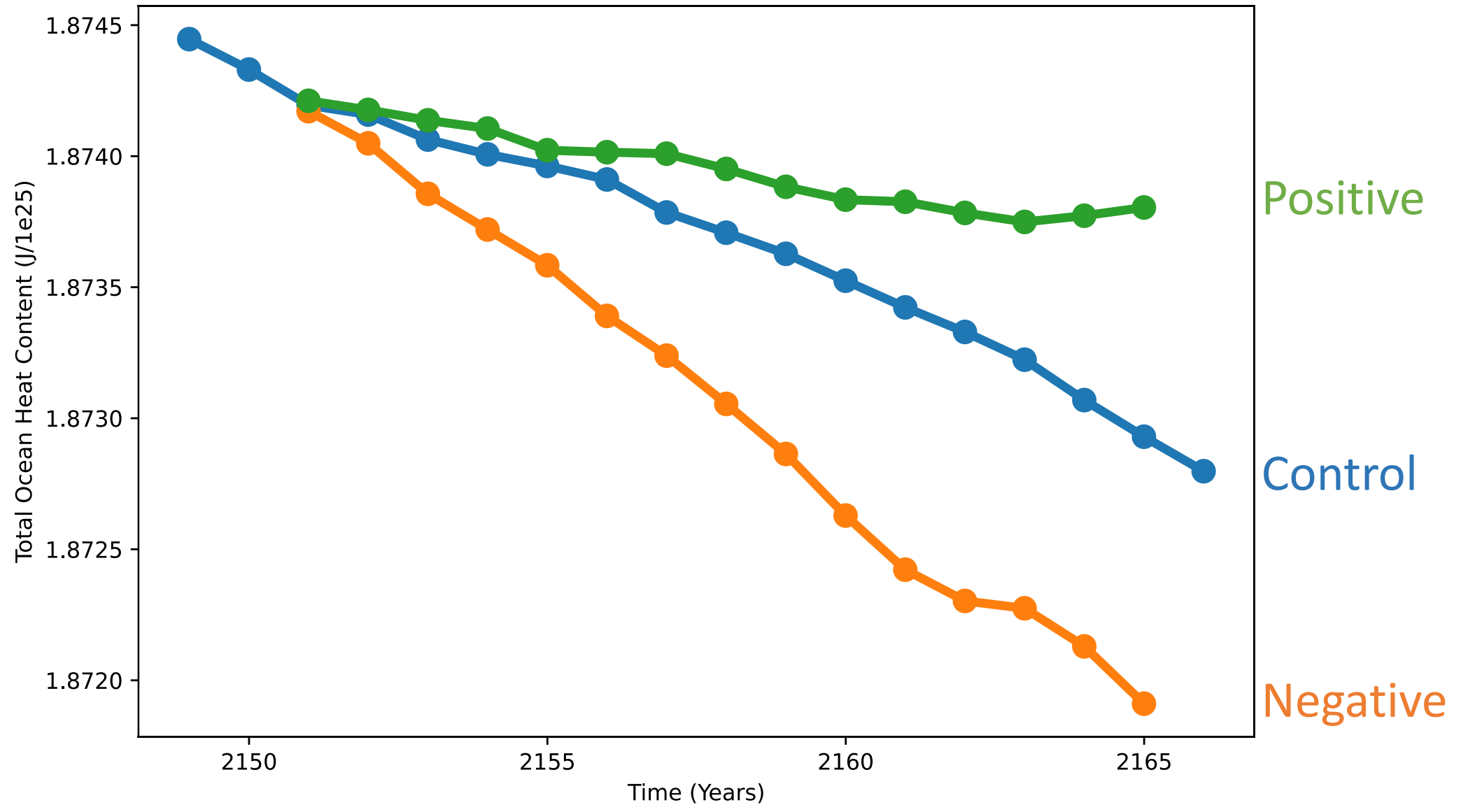
Small differences lead to rapid divergence in answers

Hence bitwise reproducibility is critical.





ACCESS-OM2-01 RYF9091 Spin-Up with Easterly Antarctic Wind Perturbations





# Create and Release Control Experiments

## Why?

- Saves human and compute resources
- Supported and tested for reproducibility
- Perturbation experiments can be compared
- Useful experiment in its own right

## Implications

- Long term support of data and model: up to 7 years!
- Scheduled testing ensures experiments still reproduce
- Test upgraded dependencies (stack) to avoid software aging



# Replicability: verify robust results

Scientific replicability: same code, same model configuration and inputs on different hardware will give **different** specific answers, but the representation of the climate will be **statistically indistinguishable**

Replicability is important: when other researchers can replicate results it implies the model is robust and this builds trust



## Conclusion

**Reliability:** it just works is foundational, researchers trust the model

**Reproducibility:** essential answers don't change. Built on the foundations of reliability

**Replicability:** verify robust results. Highly desirable, facilitated by reliability and reproducibility

Getting this right means models will be trusted, and trustworthy. This has major real-world ramifications.





## Conclusion

**Reliability:** it just works is foundational, researchers trust the model

**Reproducibility:** essential answers don't change. Built on the foundations of reliability

**Replicability:** verify robust results. Highly desirable, facilitated by reliability and reproducibility

Getting this right means models will be trusted, and trustworthy. This has major real-world ramifications.





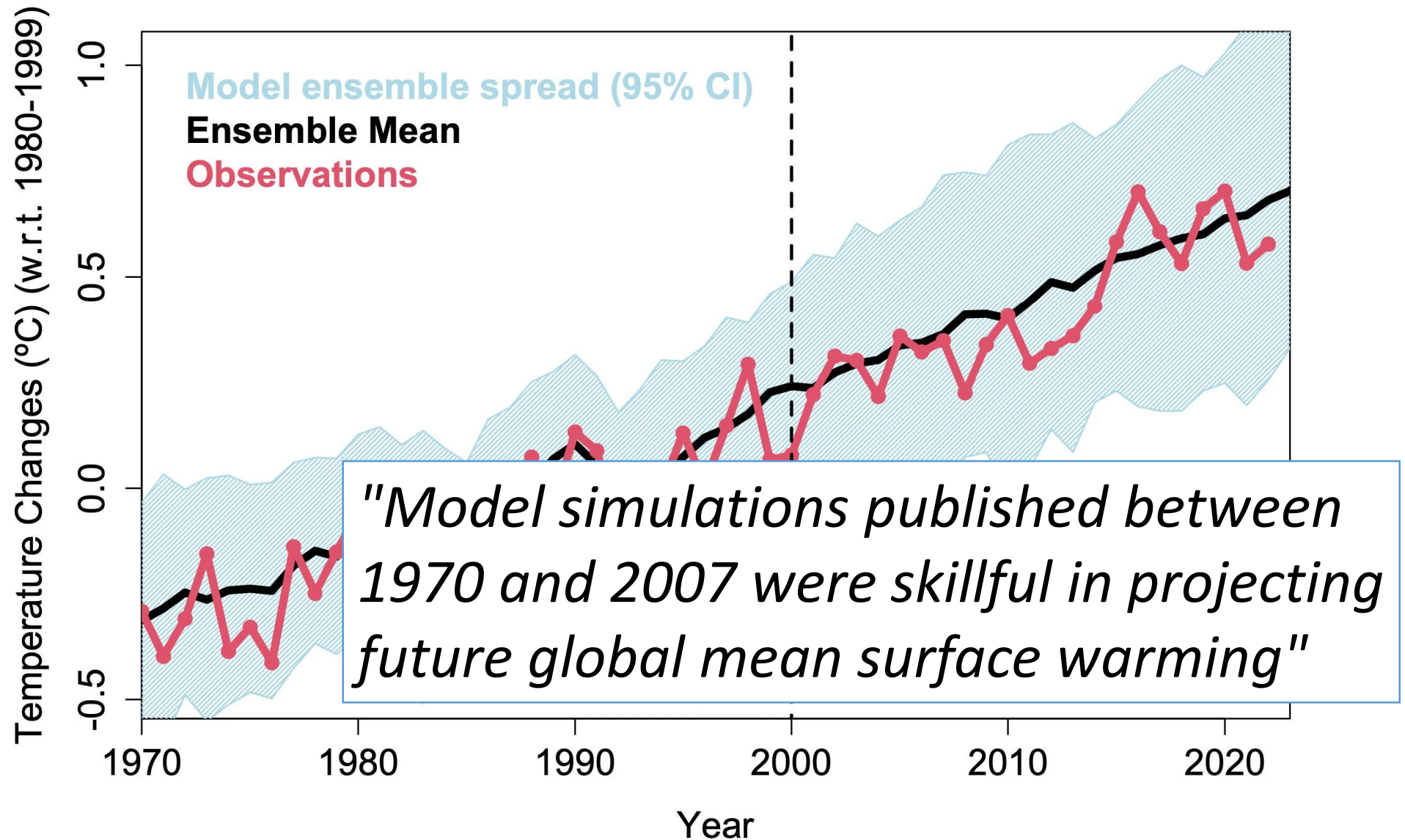
## Other testing

Build reproducibility CI – use reproducible model builds as "quick" metric for model reproducibility

Model reproducibility – test model configuration still reproduce to detect issues with underlying infrastructure

Performance testing – ensure code changes don't degrade performance

## Forecast evaluation for models run in 2004



Credit: Gavin Schmidt

<https://climate.nasa.gov/news/2943/study-confirms-climate-models-are-getting-future-warming-projections-right/>



1961

0.506127

