Workshop on Correctness and Reproducibility for Earth System Software
(in conjunction with the tutorial on Rigor and Reasoning in Research Software)

November 5-7, 2025

NSF National Center for Atmospheric Research (NCAR), Boulder, CO

Co-chairs

Alper Altuntas, CGD, NSF National Center for Atmospheric Research
Allison Baker, CISL, NSF National Center for Atmospheric Research

Committee

John Baugh, Civil Engineering and Operations Research, North Carolina State University
Ilene Carpenter, Earth Sciences Segment Manager, Hewlett Packard Enterprise
Brian Dobbins, CGD, NSF National Center for Atmospheric Research
Michael Duda, Mesoscale & Microscale Meteorology Lab, NSF National Center for Atmospheric Research
Karsten Peters-von Gehlen, Department of Data Management, Deutsches Klimarechenzentrum GmbH
Ganesh Gopalakrishnan, Kahlert School of Computing, University of Utah
Dorit Hammerling, Applied Mathematics and Statistics, Colorado School of Mines
Balwinder Singh, Atmospheric Sciences and Global Change, Pacific Northwest National Laboratory

Abstracts

Wednesday Morning Session

Ch.1: Code and Fix *(Alper Altuntas, NSF NCAR)*
- Introduction to the running example: 1-D Heat Equation
- Implement a quick solution and observe the drawbacks of unstructured, monolithic code

Ch.2: Reasoning about code *(Alper Altuntas, NSF NCAR)*
- Designing code for reasoning and testability
- Abstraction, Decomposition, Specification, Implementation

Ch.3: Unit Testing (*Manish Venumuddula, NSF NCAR***)**
- Introduction to pytest
- Writing effective unit tests for scientific code

Ch.4: Property-Based Testing (*Deepak Cherian, Earthmover***)**
- Generative testing with Hypothesis
- Catching edge cases humans wouldn't think to test.

Wednesday Afternoon Session

Ch.5: Theorem Proving *(Alper Altuntas, NSF NCAR)*
- Introduction to Z3
- Exhaustively reason about code behavior

Functional Testing (*Adrianna Foster, NSF NCAR***)**

- Reasoning about the “scientific faithfulness” of code.
- Applications: large scale, production code

Thursday Morning Session

Keynote: Lean into Verifiable Intelligence
Soonho Kong, Amazon Web Services

The convergence of artificial intelligence and formal verification is creating a transformative virtuous cycle that will reshape how we create and validate knowledge. On one side, AI systems increasingly need verification to ensure correct outputs and build trust. On the other, formal verification methods require AI to overcome fundamental challenges in scalability, usability, and automation. This bidirectional relationship is already revolutionizing mathematics through systems like Lean4, where AI assists in proof discovery while formal foundations ensure mathematical rigor, demonstrating how this synergy accelerates reliable knowledge creation. The success of Lean4 in transforming mathematical research offers a compelling blueprint for achieving verifiable intelligence across domains, fundamentally changing our approach to creating systems that are both powerful and provably correct.

Invited Talk: When Do Explanations Explain? A Controlled Study of XAI Under Varying Signal-to-Noise
Antonios Mamalakis, University of Virginia

Neural networks deliver strong predictions across the sciences, but their opacity limits adoption, especially in geoscience, where understanding why a forecast is made matters. Explainable AI (XAI) methods aim to bridge this gap, yet their outputs are method-dependent and can disagree, and even faithful attributions may be physically misleading if the underlying model has learned noise rather than signal. We conduct a controlled study using a synthetic benchmark in which the true drivers of the target are known. By training neural networks across different data sizes and target noise levels (conditions that mirror many observational Earth system datasets), we test when XAI methods recover the true explanatory structure. Two main results emerge. First, explanatory fidelity increases as models capture a larger fraction of the learnable, signal-driven variance. Second, inter-method agreement tracks this fidelity and can serve as a practical proxy when ground truth is unavailable. Conversely, in low signal-to-noise or data-scarce regimes, explanations degrade and methods diverge. These findings offer concrete guidance for deploying XAI in geosciences and beyond: prioritize models that demonstrably learn signal, and use cross-method consensus as an operational check on explanation reliability.

Invited Talk: Validating and Enforcing Physical Consistencies in AI weather prediction models
David John Gagne, NSF NCAR

AI weather prediction models have demonstrated remarkable increases in accuracy over traditional NWP models with far less latency and computational requirements for prediction. However, there are a growing number of examples where the improvements in performance come at the expense of physical consistency guarantees that are necessary assumptions for downstream applications like data assimilation. AI weather prediction models also tend to experience error growth in ways that differ noticeably from physics-based models. This presentation will examine different error scenarios for AI

weather prediction models and show how some of these errors are being mitigated through architecture and physics constraints in the NCAR CREDIT platform.

***Panel:* AI/ML Reasoning and Explainability in Scientific Computing**
Soonho Kong, Amazon Web Services
Antonios Mamalakis, University of Virginia
David John Gagne, NSF NCAR
Moderator: Dorit Hammerling, Colorado School of Mines

Thursday Afternoon Session

A Lightweight Verification Strategy of Climate Coupler Correctness
Chinmayi Baramashetru, University of Kent

Earth system models are complex systems composed of interacting subsystems e.g., including atmosphere, ocean, cryosphere, biosphere, each consisting of a broad range of physical and chemical processes that operate over vastly different spatial, temporal, and computational scales. A critical requirement for such systems is the seamless flow of data between subsystems, typically achieved through a coupler- a software component that manages interactions between submodels. Couplers perform core tasks such as data transfer and routing, grid interpolation, and time synchronization. However, their complexity and loose specification makes them a subtle but significant source of integration errors. Because couplers are widely reused across many models, any error in their implementation can propagate broadly and silently across components, undermining scientific conclusions in climate projections. In this work, we advocate for a lightweight formal verification strategy to enhance coupler correctness. We demonstrate this approach by employing a hybrid strategy that combines static and runtime techniques to enforce module-level contracts. As a case study, we apply it to a minimal yet representative kernel of the widely used OASIS-MCT coupler. The module we focus on is a core component of OASIS's data exchange mechanism and serves as a realistic target for verification. Through this case study, we show how contracts can be incrementally introduced to validate critical coupler tasks. Finally, we outline how this verification approach can be retrofitted into existing couplers or applied to future ones, offering a scalable path towards more robust, auditable, and formally verifiable climate modeling infrastructure.

Verification and validation of the next generation ocean model Omega for the Energy Exascale Earth System Model
Carolyn Begeman, Los Alamos National Laboratory

The U.S. Department of Energy's Energy Exascale Earth System Model (E3SM) was originally developed for CPU-dominated supercomputer architectures. With the growing use of GPU-dominated machines, the E3SM team recognized the opportunity to exploit these resources for eddy-resolving simulations. New atmosphere and ocean components are now being developed for E3SM in C++ with the Kokkos library. Here, I present our approach to verification and validation for this new ocean component, Omega, incorporating benchmarking against the existing E3SM ocean component, MPAS-Ocean. We have built up a suite of test cases that can be set-up and executed for either ocean component through the same Python library, Polaris. Polaris augments unit testing with bit-for-bit testing during code review in setups that target correctness, realism, and numerical stability. Polaris also has consolidated several aspects of

our workflow by offering conda environment configurations for multiple supercomputers, model build utilities, unstructured mesh creation, and visualization on the native mesh.

METplus: A one-stop shop for trusted, reproducible, and scalable verification capabilities
Michelle Harrold, NSF NCAR

The enhanced Model Evaluation Tools (METplus) system is a community-driven verification and diagnostic framework that continues evolving to meet the growing needs of the operational and research communities. METplus integrates a powerful suite of core tools, Python wrappers, and modular diagnostic components designed to evaluate model performance across a wide range of spatial and temporal scales. Its flexible, scalable, and extensible design enables users to implement traditional and advanced statistical techniques, including deterministic, probabilistic, and ensemble-based verification methods. Recent developments focus on streamlining usability, improving scalability and optimization for large ensemble datasets (both high-resolution and coarse), and expanding diagnostic capabilities tailored to the growing complexity of Earth system models. This presentation will highlight METplus's role in supporting robust and reproducible workflows through METplus use cases, with emphasis on recent enhancements.

Assessing Statistical Reproducibility of AI/ML Surrogates of Weather and Climate Models
Salil Mahajan, Oak Ridge National Laboratory

The use of AI/ML surrogates in weather and climate modeling offers the ability to generate very large ensembles at low computational cost. In contrast, traditional physics-based models are typically too expensive to support ensemble sizes of comparable scale. To enable a fair comparison, we use an ultra-low-resolution configuration of the DOE E3SMv3 model that retains the full complexity of the physical system while allowing the generation of large ensembles. In this study, we train a reduced form of FourCastNet v1—an AI-based emulator—on output from the ultra-low-resolution E3SMv3 model and assess its statistical reproducibility. We generate large ensembles using both the dynamical model and the trained surrogate under small (O(0.1) K) and large (O(1) K) initial condition perturbations. Results show that the AI surrogate exhibits significantly reduced ensemble spread under large perturbations and virtually no variability under small perturbations, indicating limited ability to replicate the internal variability captured by the dynamical model. Initial tests also show that increasing the architectural complexity of FourCastNet—such as adding more transformer layers or width—does not significantly improve ensemble reproducibility. We plan to present the evaluation of newer and more advanced surrogate architectures (e.g., later versions of FourCastNet), explore stochastic seeding methods to enhance ensemble diversity, and apply formal statistical reproducibility evaluation frameworks to assess emerging AI-based models.

MAM4xx: A Performance-Portable Aerosol Model for Next-Generation Earth System Simulations
Balwinder Singh, Pacific Northwest National Laboratory

Aerosols are integral to atmospheric processes, influencing the atmosphere both directly, through interactions with solar and terrestrial radiation, and indirectly, by modifying cloud properties and precipitation patterns. However, many global Earth system models struggle to accurately represent aerosol properties due to simplified process parameterizations and coarse spatial resolution. Conventional grid spacings (~50 km) are insufficient to resolve the fine-scale variability of aerosol distributions and their interactions with cloud systems, limiting the realism

of aerosol–cloud feedback.

With the capability of models like the Energy Exascale Earth System Model (E3SM) to run at kilometer-scale resolution, there is a growing need for prognostic aerosol models that can operate efficiently at these finer scales. To meet this need, we developed MAM4xx, a modernized and performance-portable version of the Modal Aerosol Model (MAM4). MAM4xx has been fully rewritten in C++ using the Kokkos performance portability library, enabling efficient execution on GPU-accelerated architectures.

Integrated into the E3SM's atmospheric component (EAMxx), MAM4xx supports several resolutions, including the target ~3 km convection-permitting scale. It is designed for cross-platform portability and has been successfully deployed on major leadership-class HPC systems, including OLCF's Frontier, ALCF's Aurora, and NERSC's Perlmutter, as well as institutional clusters.

This advancement enables next-generation Earth system modeling with prognostic aerosol representation at km-scale resolutions. We have rigorously validated MAM4xx against the legacy Fortran implementation, using newly developed end-to-end tests to ensure an accurate port. Continuous integration pipelines across CPU and GPU architectures enable early bug detection and safeguard reproducibility. Strict version control, standardized test suites, and cross-platform checks ensure consistent model performance across resolutions. In this presentation, we will share MAM4xx-enabled EAMxx simulation results, along with our validation strategy, performance testing insights, and lessons learned.

Directional Sign Loss: A Topology-Preserving Loss Function that Approximates the Sign of Finite Differences
Harvey Dam, University of Utah

Preserving topological features in learned latent spaces is a fundamental challenge in representation learning, particularly for topology-sensitive data. This paper introduces directional sign loss (DSL), an efficient, differentiable loss function that approximates the number of mismatches in the signs of finite differences between corresponding elements of two arrays. By penalizing discrepancies in critical points between input and reconstructed data, DSL encourages autoencoders and other learnable compressors to retain the topological features of the original data. We present the formulation and complexity analysis of DSL, comparing it to other non-differentiable topological measures.
Experiments on multidimensional array data show that combining DSL with traditional loss functions preserves topological features more effectively than traditional losses alone. DSL serves as a differentiable, efficient proxy for common topology-based metrics, enabling topological feature preservation on previously impractical problem sizes and in a wider range of gradient-based optimization frameworks.

METplus: Continuous integration with GitHub Actions for reproducible results and reduced software vulnerabilities
John Halley Gotway, NSF NCAR

METplus is a software suite used around the world to perform consistent, reproducible verification of Numerical Weather Prediction model output. It consists of multiple highly configurable software

components housed in separate GitHub repositories. Community contributions are encouraged to extend the capabilities. However, this flexibility makes effective testing challenging. Individual software components may be sufficiently tested but changes may inadvertently affect other components. Software dependencies required for one user's contributions may not be available in other users' computing environments.

The METplus team has implemented a complex continuous integration framework designed to address these challenges. METplus leverages GitHub Actions to trigger workflows that perform a variety of tasks. Over 100 use case examples are provided in the top-level METplus GitHub repository. They are automatically tested to confirm that changes to the METplus software components do not break or change the output of existing use cases. A set of rules determine what should be run during an automated workflow. A subset of tests can be run when a developer pushes changes to a repository. The full suite of tests and logic to compare the output to truth data are run when a request is made to merge changes. A developer can also manually enable or disable testing components. This flexibility improves the development process while avoiding unnecessary execution of jobs.

Through the generous support of the United States Air Force, the METplus continuous integration framework has recently been enhanced to include routine scanning for Common Vulnerabilities and Exposures (CVEs) as well as other software vulnerabilities using the SonarQube static code analysis tool.

Docker integrates nicely into GitHub Actions and is used to perform many useful functions of the METplus testing and security scanning workflows. In this talk, I will describe the METplus repository hierarchy and our use of Docker and GitHub actions to facilitate automated regression testing which streamlines the development process and saves time and money. I will discuss our recent emphasis on automated and routine scanning for CVEs and with SonarQube which provides more robust results and builds trust among both our funding partners and the broad community of METplus users.

Reproducibility of Earth System Software for Weather and Climate Analytics
Deepak Kumar, Texas Tech University

Reproducibility is a cornerstone of scientific integrity, yet achieving it remains a persistent challenge in Earth system science, where complex software tools and large-scale datasets are routinely used for weather and climate analytics. This study examines the reproducibility of Earth system software, evaluating the extent to which computational workflows in weather and climate research can be reliably replicated across platforms, environments, and user groups. We review common sources of irreproducibility, including dependency mismatches, evolving data formats, undocumented preprocessing steps, and inconsistent computational environments. Through case studies of widely used software frameworks—such as WRF, CESM, and ESMValTool—we highlight both successful reproducibility practices and recurring obstacles. We also assess emerging tools and standards aimed at enhancing reproducibility, such as containerization (e.g., Docker, Singularity), workflow automation systems, and FAIR (Findable, Accessible, Interoperable, Reusable) data principles. Our findings underscore the need for community-driven efforts to adopt transparent documentation, standardized workflows, and long-term software sustainability strategies. By addressing these challenges, the Earth system science community can build more reliable, accessible, and impactful tools for advancing climate and weather research.

Friday Morning Session

Tutorial: Git/GitHub and Automated Testing
Helen Kershaw, NSF NCAR

Compression Safeguards towards Safe and Fearless Lossy Compression
Juniper Tyree, University of Helsinki

The volume of data output by high-resolution weather and climate models is increasing faster than improvements in the methods for storing, accessing, and analysing this data, threatening to limit future model development. Lossy data compression methods sacrifice precision to reduce data size significantly. Although some lossy compressors promise compression ratios of 100x or more, the fear of losing important information and a lack of trust in lossy compression have thus far limited their use.

We present compression safeguards, which help overcome this trust gap by (i) enabling scientist users to precisely express their safety requirements, including regionally varying error bounds over the data or derived quantities of interest, and (ii) guaranteeing that these requirements are always met. The compression safeguards can be used with any existing compressor, either during compression or afterwards, allowing for easier adoption and use within different compression frameworks.

As the burden of trust in fulfilling the user's safety requirements is shifted away from specific compressor implementations towards the safeguards, even untrusted, potentially misbehaving compressors can then be used safely and without fear when combined with safeguards, which we hope will aid their adoption for scientific data.

Formal Guarantees for Error-Bounded Lossy Compression
Tripti Agarwal, University of Utah

Error-bounded lossy compression has emerged as a crucial technique for reducing the massive data volumes generated by high-performance computing (HPC) and climate simulations, enabling efficient storage and transmission while maintaining user-specified error bounds. Extending these compressors to support homomorphic operations (computations directly on compressed data) offers significant advantages but demands strong correctness guarantees. Our work addresses this challenge by introducing a structured approach to correctness, distinguishing between user-level correctness (e.g., preservation of data topology) and implementation-level correctness (e.g., absence of data races and numeric errors). We investigate the application of formal methods to establish these guarantees, beginning with low-level bug detection using data race-checking, numeric error bounding, and exception analysis. Building on these results, we explore the use of separation logic-based verification to formally reason about and ensure correctness. This research lays the groundwork for reliable, verifiable homomorphic compression libraries suitable for widespread deployment in scientific computing workflows.

High Performance Climate and Weather Benchmark (HPCW): continued development of reproducible benchmarks
Niclas Schroeter, German Climate Computing Center - DKRZ

The High Performance Climate and Weather Benchmark (HPCW) has been continually developed in multiple European projects, namely ESCAPE2, ESiWACE2 and, now, in ESiWACE3 and HANAMI. The purpose of HPCW is to isolate key aspects of climate and weather models, representing them in a domain-specific benchmark suite. This allows for a detailed performance comparison across different systems and architectures. HPCW is fully open-source and contains benchmarks from multiple European ESM models (ICON, Nemo) and mini-applications which were extracted from the IFS.

This talk acts as a follow-up to the presentation held at the inaugural Workshop on Correctness and Reproducibility for Climate and Weather Software. This year's presentation will begin with an overview of the systems behind HPCW, which allow the user to compile and run all applications, alongside their dependencies if necessary, while only requiring very little information from the user. We will then go into detail regarding the changes that happened over the last two years, including the consolidation of the HPCW components and the move to a fully open-source framework. We will talk about the experience that we gained from using HPCW on multiple different systems and the ongoing projects that use HPCW as a benchmark vehicle, all of which has the goal to further improve both the hardware and software landscape for the ESM community.

Assessing the quality of zarr datasets from Km scale ESM simulations
Kameswar Rao Modali, German Climate Computing Center - DKRZ

As the exascale computers are becoming the workhorses in HPC centers across the globe, the Earth System models are gearing towards Km scale simulations to scale up to the enhanced computing capabilities. In order to enable the federated access of the output from these Km scale simulations to researchers across the globe, a paradigm shift is needed in the way the output from these simulations is stored as well as accessed. In this direction, the ESMs are adopting cloud native storage formats like zarr in tandem with interpolation onto healpix grids, for their output. As these are well suited for ‘larger than memory’ analysis tasks of the scientific workflows. However, the quality control of these datasets in such formats and at that scale is yet to be explored thoroughly. In this work, we address this aspect and provide some insights into some of the possible checks that could ascertain the quality of the data as well as the simulations.

Enhancing Scientific Reproducibility in CrocoDash for Regional Ocean Modeling
Andrew Kwong, Cornell University

Regional ocean models provide powerful tools for studying the impacts of climate change at finer spatial scales. By simulating a subset of the global ocean, they offer insights into physical and biogeochemical processes that are often unresolved in global climate models. This work incorporates novel scientific reproducibility and software correctness features into CrocoDash, which is a Python package for rapidly prototyping regional Modular Ocean Model 6 (MOM6) cases within CESM. The new features include enhanced configurators for grid generation and CESM case creation with built-in support for Git-based version control, snapshotting, and edit history tracking. These capabilities not only streamline case setup but also lay the foundation for systematic scientific validation, enabling more transparent, reproducible, and hypothesis-driven Earth system modeling.

Friday Afternoon Session

Characterizing and Correcting Non-Standard Arithmetic in GPUs for Scientific Computing
Paul Jiang, Purdue University

There is increasing interest in the use of GPU-based hardware accelerators in computationally demanding applications such as climate and weather simulation. Unfortunately, their non-standard arithmetic that deviates from the IEEE 754 standard results in inconsistent output when code is ported across them. Recent research has successfully characterized the architectural properties of these units, such as their block widths, padding bits, and rounding modes. Despite this, a critical gap remains in understanding their precise emergent numerical behaviors across multiple iterations as well as ensembles of data. This work builds directly on recently formalized SMT theories of these accelerators extended to consider modern GPUs and studies the behaviors that may deviate beyond matrix multiplications. We present techniques to mitigate these numerical inconsistencies as well the computational costs with which these techniques are obtainable.

A short history of artificial intelligence and what it can teach us
Anissa Zacharias, NSF NCAR

The term “artificial intelligence” has become commonplace today, but what, specifically, are we invoking when we categorize something as “AI”? From more traditional machine learning techniques to viral deepfakes, what we lexiconically label as “AI” is detrimentally broad.Our conceptions (and misconceptions) of “thinking machines” affect how we use these tools.

From the 1955 Dartmouth Summer Research Project on Artificial Intelligence that popularization of the term and the subsequent disagreements about what AI as a field would be, through the contentious Lighthill Report and the following AI “winters” of the 1970s and 1990s, we can follow iterations of our modern conversations about what AI is and what computers can (and cannot) do. In this talk, I will discuss the origins of the field of artificial intelligence, trace its path to the present day, and explain how that human-led journey influences conversations about correctness, reproducibility, and verification today.

The Fused Multiply-Add and Global Atmospheric Models: An Investigation into a Surprising Correctness Scenario
Teo Price-Broncucia, NSF NCAR