Report on the Rigor and Reasoning in Research Software (R3Sw) Tutorial *in conjunction with* the 2nd Workshop on Correctness and Reproducibility for Earth System Software

Alper Altuntas[1,2] and Allison Baker[1]

[1]US National Science Foundation National Center for Atmospheric Research (NSF NCAR), Boulder, CO, USA
[2]2025 Better Scientific Software (BSSw) Fellow

February 12, 2026

Contents

1 Introduction **4**

2 Motivation **4**

3 Event Overview **5**

3.1 Tutorial: Rigor and Reasoning in Research Software 5

3.2 Special Session: Verifiable and Explainable AI/ML in Scientific Computing 7

3.3 Workshop: Correctness and Reproducibility for Earth System Software 9

4 Organization and Logistics **9**

4.1 Organizing Team 10

4.2 Attendance 10

4.3 Post-Event Survey 11

4.4 Event Logistics 11

5 Conclusion **11**

5.1 Acknowledgment 12

A Links and Resources **12**

B Event Program **12**

C Abstracts **15**

C.1 Lean into Verifiable Intelligence 15

C.2 When Do Explanations Explain? A Controlled Study of XAI Under Varying Signal-to-Noise 15

C.3 Validating and Enforcing Physical Consistencies in AI weather prediction models . . 16

C.4 A Lightweight Verification Strategy of Climate Coupler Correctness 16

C.5 Verification and validation of the next generation ocean model Omega for E3SM . . 17

C.6 METplus: A one-stop shop for trusted, reproducible, and scalable verification capabilities 17

C.7 Assessing Statistical Reproducibility of AI/ML Surrogates of Weather and Climate Models 18

C.8 MAM4xx: A Performance-Portable Aerosol Model for Next-Gen Earth System Simulations 18

C.9 Directional Sign Loss: A Topology-Preserving Loss Function that Approximates the Sign of Finite Differences 19

C.10 METplus: Continuous integration with GitHub Actions for reproducible results and reduced software vulnerabilities 19

C.11 Reproducibility of Earth System Software for Weather and Climate Analytics 20
C.12 Compression Safeguards towards Safe and Fearless Lossy Compression 21
C.13 Formal Guarantees for Error-Bounded Lossy Compression 21
C.14 High Performance Climate and Weather Benchmark (HPCW) 22
C.15 Assessing the quality of zarr datasets from Km scale ESM simulations 22
C.16 Enhancing Scientific Reproducibility in CrocoDash for Regional Ocean Modeling . . 22
C.17 Characterizing and Correcting Non-Standard Arithmetic in GPUs for Scientific Computing 23
C.18 A short history of artificial intelligence and what it can teach us 23

References **24**
Workshop Talks 24

1 Introduction

The "Rigor and Reasoning in Research Software (R3Sw) Tutorial," held in conjunction with the "2nd Workshop on Correctness and Reproducibility for Earth System Software," took place on November 5–7 at the Mesa Laboratory of the NSF National Center for Atmospheric Research (NCAR) in Boulder, Colorado (Figure 1). The event drew a total of 103 registrants, including 47 in-person participants and 56 virtual attendees.

The tutorial provided a series of hands-on lectures and interactive sessions focused on practical techniques for making scientific software more reliable, understandable, and trustworthy. The core tutorial program was delivered on the first day, complemented by follow-up sessions on days two and three. The topics included unit testing, property-based testing, formal verification, functional scientific testing, verifiable and explainable AI, and automated testing with GitHub, featuring invited speakers from academia and industry.

The workshop sessions featured 16 contributed talks covering verification of core model components, performance-portable model development, reproducible workflows, and the increasing role of AI in Earth system science. Presentations demonstrated advances in the application of formal methods, continuous integration, lossy-compression safeguards, benchmark suites, and reproducible model-configuration tools. Together, the sessions highlighted a community-wide effort to strengthen confidence in Earth system simulations and data-driven scientific computing.

As a whole, the tutorial and the workshop provided a dedicated forum for Earth system modelers, software engineers, and the broader scientific software community to discuss challenges, opportunities, and recent advances in ensuring software correctness and reproducibility.

This event was primarily sponsored by the Better Scientific Software (BSSw) Fellowship, with additional support from the CGD and CISL Laboratories at NSF NCAR.

2 Motivation

Scientific computing applications, such as earth system models, are the result of many years or even decades of development, resulting in large and continually evolving code bases [1, 2, 3]. These sophisticated software systems must constantly adapt to emerging scientific questions, as well as to rapidly changing high-performance computing (HPC) architectures [4]. At the same time, their outputs support critical decisions in areas such as water resources, agriculture, public health, and responses to extreme events [2, 5]. Maintaining confidence in these models, and in the scientific software ecosystem that supports them, requires sustained attention to correctness, robustness, and reproducibility.

Despite their importance, much scientific software is still developed in a "code-and-fix" style [6], with domain scientists prototyping quickly, patching issues as they arise, and relying on bitwise-comparison tests as the primary form of regression evaluation [1]. While efficient for rapid scientific exploration, this workflow provides limited assurance that the software behaves as intended, particularly as codebases grow more complex over time. For chaotic applications like earth system models, even roundoff-level perturbations (due to compiler changes, optimization choices, or hardware differences) can lead to divergent simulation trajectories. Distinguishing harmless numerical variability from genuine software errors is therefore an ongoing challenge [7, 8].

In addition to these challenges, the scientific software stack now includes workflow managers, analysis pipelines, data assimilation systems, and increasingly, machine-learning (ML) components. ML-based surrogates and forecast models promise substantial gains in accuracy and computational efficiency, but they also raise new questions about physical consistency, statistical reproducibility, and interpretability [9, 10]. Traditional testing approaches focused on bitwise identical results or limited end-to-end experiments are insufficient for evaluating such systems in a streamlined way.

The R3Sw Tutorial and the Workshop on Correctness and Reproducibility for Earth System Software were organized in response to these intertwined challenges. The tutorial introduced a "ladder of rigor" for scientific software, from conventional unit tests to property-based testing and formal verification, emphasizing a scientific mindset toward software development in which behavioral expectations are specified, tested, challenged, and refined. The workshop complemented this perspective with a diverse set of case studies and community efforts from across the Earth system modeling community, highlighting practical methods for software verification, statistical consistency testing, workflow reproducibility, and data compression safeguards, as summarized in Section 3.3. Together, these efforts aim to strengthen confidence in scientific results by ensuring that the software used to generate them is reliable, robust, and reproducible.

3 Event Overview

This section summarizes the three-day program and clarifies how the narrative components used in this report map onto the agenda in Appendix B. Here, the term *tutorial* refers to the R3Sw hands-on instructional material delivered primarily on Day 1, with a short follow-up tutorial session on Day 3 (morning). The *special session* refers to the invited session on Verifiable and Explainable AI/ML, which occupied the Day 2 morning block. The remaining contributed-talk programming (Day 2 afternoon and Day 3 after the follow-up tutorial) constitutes the *workshop* component. Together, these pieces formed a progression from methods and tooling (tutorial), to an emerging application area and community discussion (special session), to community case studies and practice (workshop).

3.1 Tutorial: Rigor and Reasoning in Research Software

In the program agenda (Appendix B), this tutorial component corresponds to the full Day 1 instructional program and the brief follow-up tutorial session on Day 3 (morning) on Git/GitHub and automated testing. These sessions were aimed at a broad scientific computing audience and were organized around hands-on, example-driven activities.

Scientific software plays a central role in modern research, from weather prediction to molecular modeling, data assimilation, and AI-driven discovery. Yet much of this software is still developed in a "code-and-fix" style, with rapid prototyping followed by incremental patching. While this approach supports scientific creativity and speed, it often leads to code that is fragile, difficult to test, and challenging to extend when new research questions arise [11].

The R3Sw tutorial aimed to break this cycle by drawing inspiration from the scientific method itself. Just as scientific inquiry progresses through hypotheses, tests, refutation, and refinement, robust scientific software can be developed through a similar process of specifying behavior and expectations, checking them systematically, and iteratively improving both the code and our understanding

Figure 1: The "Rigor and Reasoning in Research Software (R3Sw) Tutorial," held in conjunction with the "2nd Workshop on Correctness and Reproducibility for Earth System Software," took place on November 5–7 at the Mesa Laboratory of the NSF National Center for Atmospheric Research (NCAR) in Boulder, Colorado.

of the underlying concepts being modeled. This framing provided the thematic foundation for the entire tutorial.

To make these ideas concrete, the tutorial followed a running example: the progressive development of a simple one–dimensional heat equation solver. The example began as an ad-hoc, monolithic script that resembling many early-stage research prototypes, and was gradually transformed into modular, testable, and well-specified software. Along the way, participants gained practical experience with a structured "ladder of rigor," consisting of:

- **Unit testing with `pytest`**, establishing a baseline of confidence in code behavior.
- **Property-based testing with `Hypothesis`**, exploring input spaces more broadly and revealing edge cases.
- **Theorem proving with `Z3py`**, enabling exhaustive reasoning and strong guarantees for selected, critical routines.
- **Automated testing with GitHub Actions**, demonstrating continuous integration workflows that ensure ongoing correctness as code evolves.
- **Verifiable and Explainable AI/ML in Scientific Computing**, a special session featuring invited talks and a panel discussion on emerging methods for trustworthy data-driven modeling. (See Section 3.2.)

Each step offered new capabilities, tradeoffs, and insights into the design of research software. Collectively, these techniques emphasized a shared goal: bringing a scientific mindset to software development. Participants were encouraged to articulate desired code properties, express them

Figure 2: Guest lecturer Deepak Cherian (Earthmover) delivering a hands-on session on property-based testing during the Rigor and Reasoning in Research Software (R3Sw) Tutorial.

through specifications and invariants, and challenge them through tests and formal reasoning. All tutorial materials are available at: https://www.alperaltuntas.com/R3Sw

The tutorial was designed for a broad audience of scientists, engineers, and students across all domains of computational research. No prior experience with testing or verification was required beyond basic familiarity with Python. This inclusive structure supported participants at multiple levels of experience, from those encountering testing frameworks for the first time to experts interested in verification methods.

The tutorial was led by Alper Altuntas (NSF NCAR), who delivered the introductory and core tutorial lecture on abstraction, specification, and code design for robustness, as well as the lecture on formal verification with `z3py`. Additional talks and lectures were provided by invited speakers and guest lecturers from research institutions, industry, and academia (Figure 2). Guest lecturers and invited speakers included Soonho Kong (Amazon Web Services), Deepak Cherian (Earthmover), Antonios Mamalakis (University of Virginia), Helen Kershaw (NSF NCAR), Manish Venumuddula (NSF NCAR), and Adrianna Foster (NSF NCAR).

3.2 Special Session: Verifiable and Explainable AI/ML in Scientific Computing

In the program agenda (Appendix B), the special session corresponds to the Day 2 morning invited block.

This special session explored the rapidly evolving intersection of artificial intelligence, scientific computing, and formal reasoning, an area that is reshaping how computational applications are developed, validated, and reasoned about. The session brought together perspectives from industry and academia via invited talks, and a panel discussion to examine how AI/ML models can be made more trustworthy, physically consistent, and explainable, and how verification methods can scale to

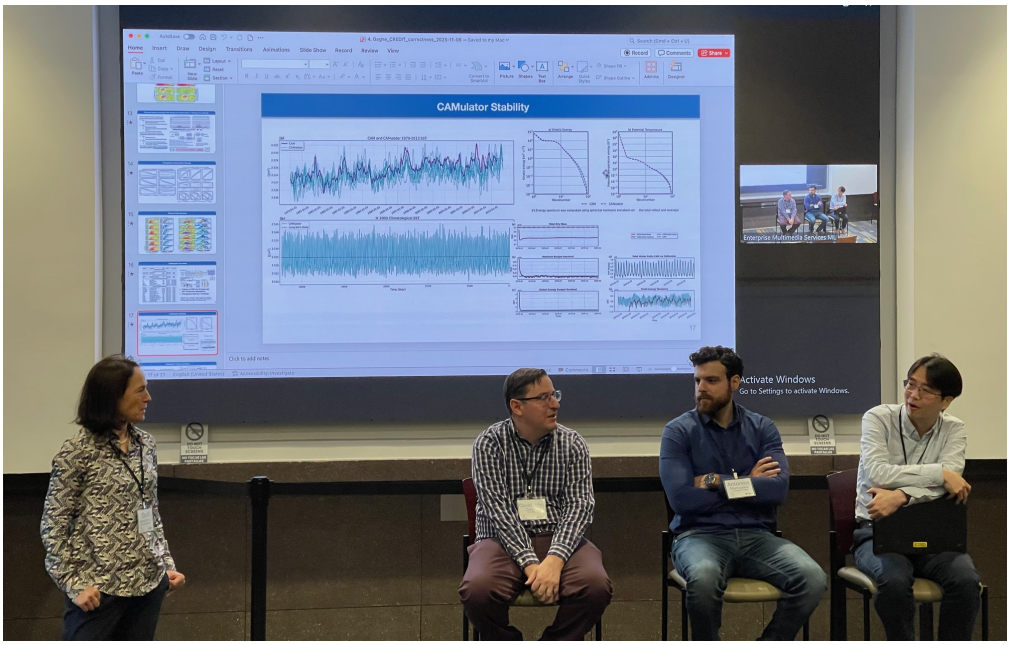

Figure 3: Panel on "Verifiable and Explainable AI". Moderator: Dorit Hammerling, Panelists: David John Gagne, Antonios Mamalakis, Soonho Kong.

meet the demands of modern AI systems.

The keynote, Lean into Verifiable Intelligence, delivered by Soonho Kong (Amazon Web Services), highlighted the emerging "virtuous cycle" between AI and formal verification. Kong emphasized that AI systems increasingly require verification to ensure correctness and reliability, while verification frameworks themselves are beginning to depend on AI to overcome longstanding barriers in automation and usability. Based on using Lean4 theorem prover [12], he demonstrated how AI-assisted proof generation and formally rigorous foundations can jointly accelerate discovery and provide a blueprint for developing powerful, verifiably correct AI systems across domains.

Invited talks expanded on challenges specific to Earth system science. Antonios Mamalakis (University of Virginia) presented a controlled study showing that explainable AI (XAI) methods are most reliable when models learn meaningful signal, and that cross-method agreement can serve as a practical proxy for explanatory fidelity. David John Gagne (NSF NCAR) discussed physical consistency in AI weather prediction models, illustrating error modes that arise when models diverge from physical constraints and showing how the NCAR CREDIT platform incorporates architectural and physics-informed strategies to mitigate them.

The session concluded with a panel discussion (Figure 3) featuring Kong, Mamalakis, and Gagne, moderated by Dorit Hammerling (Colorado School of Mines). One of the themes of the discussion was how theorem proving can be meaningfully integrated into AI/ML applications, including the roles of interactive proof systems versus automated theorem proving. The panel emphasized the importance of rigorous evaluation, hybrid physics–AI approaches, and verification-aware workflows as AI becomes increasingly embedded in scientific modeling pipelines.

3.3 Workshop: Correctness and Reproducibility for Earth System Software

In the program agenda (Appendix B), the workshop component corresponds to the contributed talks on Day 2 (afternoon) and the remainder of Day 3 following the tutorial follow-up.

The workshop featured 16 contributed talks spanning a broad range of efforts to improve the reliability and reproducibility of Earth system software and, more broadly, scientific computing applications. Several talks focused on verification and correctness in core model components, including lightweight formal verification for Earth system couplers [13], validation strategies for next-generation ocean models [14], and performance-portable aerosol modeling [15]. Community tools such as METplus [16, 17] demonstrated how continuous integration, standardized test suites, and consistent workflows can strengthen model evaluation across architectures. Other contributions examined the increasing role of AI and data-driven approaches in Earth system science, from assessing statistical reproducibility of AI surrogates [18] to developing topology-preserving loss functions [19] and evaluating reproducibility practices across widely used model analysis frameworks [20].

The Friday sessions extended these themes into data compression, workflow reproducibility, and hardware behavior. Presentations on lossy compression safeguards [21] and formal guarantees [22] emphasized the need for trustworthy compression at exascale. Talks on the HPCW benchmark suite [23], the quality of cloud-native zarr datasets for kilometer-scale simulations [24], and reproducibility enhancements in CrocoDash [25] highlighted how tools and formats are evolving to support consistent experiments and analysis across systems and institutions. The program concluded with discussions of GPU arithmetic and floating-point correctness [26, 27] and a historical perspective on artificial intelligence [28]. Together, the sessions reflected a community investing across multiple layers of the software stack to strengthen confidence in Earth system simulations and data-driven scientific computing.

4 Organization and Logistics

Planning for the R3Sw Tutorial and the 2nd Workshop on Correctness and Reproducibility for Earth System Software began in early 2025 and involved close coordination among the co-chairs, organizing committee, and administrative staff across CGD and CISL. The event was held in a hybrid format, with all sessions streamed live via Zoom to support remote participation. Recordings are publicly available on YouTube (Day 1: `https://www.youtube.com/watch?v=dCDij293s74`; Day 2: `https://www.youtube.com/watch?v=5q8ZSPDMYsE`; Day 3: `https://www.youtube.com/watch?v=wl6ADk285wc`). All tutorial materials and workshop slides are currently available via the event webpage: https://ncar.github.io/correctness-workshop/

The tutorial program and lecture materials [11], which will remain available online, were primarily developed by the tutorial lead, Alper Altuntas, with feedback and contributions from guest lecturers and members of the organizing committee.

To ensure broad community engagement, the event was advertised through a wide range of channels, including announcements to previous attendees, outreach from invited speakers and committee members, and postings across multiple UCAR and NCAR communication platforms such as SunDog, BlueSky, DailyB, UCAR Bulletin Boards, UCAR Currents, UCAR Commons, the CISL Events website, and NCAR Slack spaces. Additional advertisement reached the broader scientific software community through the WRF/MPAS Workshop, Community Earth System Model (CESM) and

The UCAR Software Engineering Assembly (SEA) mailing lists, United States Research Software Engineer Association (US-RSE) Conference, and the Better Scientific Software (BSSw) webpage, as well as LinkedIn and other communication outlets.

4.1 Organizing Team

Co-Chairs

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

Organizing 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, MMM, NSF National Center for Atmospheric Research
- Karsten Peters-von Gehlen, Department of Data Management, Deutsches Klimarechenzentrum (DKRZ)
- 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

Administrative Support

- Samantha Scalice, CISL, NSF National Center for Atmospheric Research
- Elizabeth Faircloth, CGD, NSF National Center for Atmospheric Research
- Teresa Walz, CGD, NSF National Center for Atmospheric Research

4.2 Attendance

The event drew 103 registrants, including 47 in-person participants and 56 virtual attendees. Participants represented a wide range of career stages and professional backgrounds, from graduate students to senior researchers across universities, national laboratories, research institutions, and industry, both within the United States and internationally.

4.3 Post-Event Survey

To help evaluate the event format and identify areas for improvement, the organizers distributed an anonymous post-event survey. A total of 28 responses were received (15 in-person, 12 virtual, and 1 hybrid). Overall satisfaction was high, with an average overall rating of 4.46/5 (median 4.5; 14 respondents rated the event 5/5 and 13 rated it 4/5). Respondents reported high likelihood of attending a similar future event (average 4.54/5) and recommending the event to colleagues (average 4.64/5). Satisfaction with speakers was uniformly positive (16 *Very Satisfied*, 12 *Satisfied*), with similarly strong satisfaction reported for the session times and dates (only one response in each category indicated mixed satisfaction).

Qualitative feedback highlighted several recurring themes:

- Opportunities to apply methods to participants' own codes (e.g., small-group work or a hackathon-style component).
- Interest in additional HPC- and performance-oriented topics.
- Suggestions to adjust or condense the multi-day format, depending on participant availability.

4.4 Event Logistics

The event included morning and afternoon breaks each day, with refreshments provided during 30-minute intervals. Friday's lunch was provided as part of the in-person registration. Travel support was provided to six speakers and lecturers to ensure broad participation from academia and industry.

Key dates included the abstract submission deadline on July 25, 2025; notification of acceptance on August 22, 2025; registration deadlines of October 24 (in-person) and October 31 (virtual); and the workshop and tutorial held November 5–7, 2025.

5 Conclusion

The R3Sw Tutorial and the 2nd Workshop on Correctness and Reproducibility for Earth System Software brought together researchers, software engineers, and practitioners committed to strengthening the rigor, reasoning, software correctness, and reproducibility of scientific computing. Through hands-on lectures, invited talks, contributed presentations, and panel discussions, participants engaged with a wide range of techniques spanning testing, verification, workflow reproducibility, data integrity, and dependable AI/ML that collectively support more robust scientific computing

Across the three-day program, a consistent theme emerged: the need for communities to adopt methodologies that treat scientific software with the same care and structure as scientific inquiry itself. The discussions and demonstrations showcased not only meaningful progress but also shared challenges and opportunities for future collaboration. The organizers hope that the momentum generated through this event will continue to support the development of reliable, reproducible, and well-reasoned scientific software within the broader research community.

5.1 Acknowledgment

This work was supported by the Better Scientific Software Fellowship Program, a collaborative effort of the U.S. Department of Energy (DOE), Office of Advanced Scientific Computing Research via ANL under Contract DE-AC02-06CH11357 and the National Nuclear Security Administration Advanced Simulation and Computing Program via LLNL under Contract DE-AC52-07NA27344; and by the National Science Foundation (NSF) via SHI under Grant No. 2435328.

This material is based upon work supported by the U.S. National Science Foundation National Center for Atmospheric Research, which is a major facility sponsored by the U.S. National Science Foundation under Cooperative Agreement No. 1755088. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the U.S. National Science Foundation.

A Links and Resources

- Event website: https://ncar.github.io/correctness-workshop/
- Slides: available via above website link
- Tutorial materials:
  - Lectures: https://alperaltuntas.com/R3Sw
  - GitHub Repository: https://github.com/alperaltuntas/R3Sw
- Recordings:
  - Day 1: `https://www.youtube.com/watch?v=dCDij293s74`
  - Day 2: `https://www.youtube.com/watch?v=5q8ZSPDMYsE`
  - Day 3: `https://www.youtube.com/watch?v=wl6ADk285wc`
- Survey link: https://www.surveymonkey.com/r/RFQYLG6

B Event Program

Wednesday, 11/5/2025

8:50 AM	**R3Sw Tutorial – Opening Remarks and Logistics** Alper Altuntas, NSF NCAR
9:00 AM	**Rigor and Reasoning in Research Software** Alper Altuntas, NSF NCAR
9:45 AM	**Unit Testing** Manish Venumuddula, NSF NCAR
10:30 AM	Break (snacks and beverages provided)
11:00 AM	**Property-Based Testing**

Deepak Cherian, Earthmover

12:20 PM Lunch

1:20 PM **Theorem Proving (Bounded Formal Verification)**
Alper Altuntas, NSF NCAR

2:30 PM Break (snacks and beverages provided)

3:00 PM **Functional Testing & Practical Session**
Adrianna Foster, NSF NCAR

4:30 PM Closing Remarks

Thursday, 11/6/2025

Session Moderators:
Morning: Alper Altuntas
Afternoon 1: Allison Baker
Afternoon 2: Ganesh Gopalakrishnan

8:45 AM **Opening Remarks**
Alper Altuntas, Allison Baker (NSF NCAR)

9:00 AM **Keynote: Lean into Verifiable Intelligence**
Soonho Kong, Amazon Web Services

9:50 AM **Invited Talk: When Do Explanations Explain? A Controlled Study of XAI Under Varying Signal-to-Noise**
Antonios Mamalakis, University of Virginia

10:25 AM Break (snacks and beverages provided)

10:55 AM **Invited Talk: Validating and Enforcing Physical Consistencies in AI weather prediction models**
David John Gagne, NSF NCAR

11:30 AM **Panel: AI/ML Reasoning and Explainability in Scientific Computing**
Moderator: Dorit Hammerling
Panelists: Soonho Kong, Antonios Mamalakis, David John Gagne

12:30 PM Lunch

1:30 PM **A Lightweight Verification Strategy of Climate Coupler Correctness**
Chinmayi Baramashetru*, University of Kent

1:50 PM **Verification and Validation of the Next Generation Ocean Model Omega for E3SM**
Carolyn Begeman*, LANL

2:10 PM **METplus: A One-Stop Shop for Trusted, Reproducible, and Scalable Verification Capabilities**
Michelle Harrold, NSF NCAR

2:30 PM **Assessing Statistical Reproducibility of AI/ML Surrogates of Weather and Climate Models**

Salil Mahajan*, ORNL

2:50 PM Break (snacks and beverages provided)

3:20 PM **MAM4xx: A Performance-Portable Aerosol Model for Next-Generation Earth System Simulations**
Balwinder Singh, PNNL

3:40 PM **Directional Sign Loss: A Topology-Preserving Loss Function That Approximates the Sign of Finite Differences**
Harvey Dam, University of Utah

4:00 PM **METplus: Continuous Integration with GitHub Actions for Reproducible Results and Reduced Software Vulnerabilities**
John Halley Gotway, NSF NCAR

4:20 PM **Reproducibility of Earth System Software for Weather and Climate Analytics**
Deepak Kumar, Texas Tech University

4:40 PM Optional Happy Hour (off-site): Southern Sun Pub & Brewery

*remote speaker

Friday, 11/7/2025

Session Moderators:
Morning 1: Allison Baker
Morning 2: Brian Dobbins
Afternoon: Balwinder Singh

9:00 AM **Tutorial: Git/GitHub and Automated Testing**
Helen Kershaw, NSF NCAR

9:50 AM **Compression Safeguards toward Safe and Fearless Lossy Compression**
Juniper Tyree*, University of Helsinki

10:10 AM Break (snacks and beverages provided)

10:40 AM **Formal Guarantees for Error-Bounded Lossy Compression**
Tripti Agarwal, University of Utah

11:00 AM **High Performance Climate and Weather Benchmark (HPCW): Continued Development of Reproducible Benchmarks**
Niclas Schroeter*, DKRZ

11:20 AM **Assessing the quality of zarr datasets from km scale ESM simulations**
Kameswar Rao Modali*, DKRZ

11:40 AM **Enhancing Scientific Reproducibility in CrocoDash for Regional Ocean Modeling**
Andrew Kwong*, Cornell University

12:00 PM Lunch (provided)

1:20 PM **Characterizing and Correcting Non-Standard Arithmetic in GPUs**
Paul Jiang, Purdue University

1:40 PM **A Short History of Artificial Intelligence and What It Can Teach Us**
Anissa Zacharias*, NSF NCAR

2:00 PM **The Fused Multiply-Add and Global Atmospheric Models: An Investigation into a Surprising Correctness Scenario**
Teo Price-Broncucia, NSF NCAR

2:20 PM Closing Discussion
Alper Altuntas, Allison Baker, NSF NCAR

3:30 PM Adjourn

*remote speaker

C Abstracts

Lean into Verifiable Intelligence

Soonho Kong, Amazon Web Services (Keynote)

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.

When Do Explanations Explain? A Controlled Study of XAI Under Varying Signal-to-Noise

Antonios Mamalakis, University of Virginia (Invited Talk)

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.

Validating and Enforcing Physical Consistencies in AI weather prediction models

David John Gagne, NSF NCAR (Invited Talk)

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.

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.

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.

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.

References

[1] Alper Altuntas et al. *Report on the 2023 Workshop on Correctness and Reproducibility for Climate and Weather Software*. NCAR Technical Note NCAR/TN-582+PROC. National Center for Atmospheric Research, 2024.

[2] Steve M Easterbrook et al. "Guest Editors' Introduction: Climate Change-Science and Software". In: *IEEE Software* 28.6 (2011), pp. 32–35.

[3] T. Clune and R. Rood. "Software Testing and Verification in Climate Model Development". In: *IEEE Software* 28.6 (2011), pp. 49–55. DOI: `http://dx.doi.org/10.1109/MS.2011.117`.

[4] Peter Bauer et al. "The digital revolution of Earth-system science". In: *Nature Computational Science* 1.2 (2021), pp. 104–113.

[5] Peter Bauer, Alan Thorpe, and Gilbert Brunet. "The quiet revolution of numerical weather prediction". In: *Nature* 525.7567 (2015), pp. 47–55.

[6] Tim Storer. "Bridging the chasm: A survey of software engineering practice in scientific programming". In: *ACM Computing Surveys (CSUR)* 50.4 (2017), pp. 1–32.

[7] DJ Milroy et al. *Nine time steps: ultra-fast statistical consistency testing of the Community Earth System Model (pyCECT v3. 0), Geosci. Model Dev., 11, 697–711*. 2018.

[8] Salil Mahajan et al. "A multivariate approach to ensure statistical reproducibility of climate model simulations". In: *Proceedings of the Platform for Advanced Scientific Computing Conference*. 2019, pp. 1–10.

[9] Peter D Dueben et al. "Challenges and benchmark datasets for machine learning in the atmospheric sciences: Definition, status, and outlook". In: *Artificial Intelligence for the Earth Systems* 1.3 (2022), e210002.

[10] Zied Ben Bouallegue et al. "The rise of data-driven weather forecasting: A first statistical assessment of machine learning–based weather forecasts in an operational-like context". In: *Bulletin of the American Meteorological Society* 105.6 (2024), E864–E883.

[11] Alper Altuntas et al. *Rigor and Reasoning in Research Software (R3Sw) Tutorial*. `https://www.alperaltuntas.com/R3Sw`. Accessed: 2025-02-01. 2025.

[12] Leonardo de Moura and Sebastian Ullrich. "The lean 4 theorem prover and programming language". In: *International Conference on Automated Deduction*. Springer. 2021, pp. 625–635.

Workshop Talks

[13] Chinmayi Baramashetru. "A Lightweight Verification Strategy of Climate Coupler Correctness". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[14] Carolyn Begeman. "Verification and Validation of the Next Generation Ocean Model Omega for the Energy Exascale Earth System Model". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[15] Balwinder Singh. "MAM4xx: A Performance-Portable Aerosol Model for Next-Generation Earth System Simulations". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[16] Michelle Harrold. "METplus: A One-Stop Shop for Trusted, Reproducible, and Scalable Verification Capabilities". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[17] John Halley Gotway. "METplus: Continuous Integration with GitHub Actions for Reproducible Results and Reduced Software Vulnerabilities". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[18] Salil Mahajan. "Assessing Statistical Reproducibility of AI/ML Surrogates of Weather and Climate Models". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[19] Harvey Dam. "Directional Sign Loss: A Topology-Preserving Loss Function that Approximates the Sign of Finite Differences". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[20] Deepak Kumar. "Reproducibility of Earth System Software for Weather and Climate Analytics". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[21] Juniper Tyree. "Compression Safeguards towards Safe and Fearless Lossy Compression". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[22] Tripti Agarwal. "Formal Guarantees for Error-Bounded Lossy Compression". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[23] Niclas Schroeter. "High Performance Climate and Weather Benchmark (HPCW): Continued Development of Reproducible Benchmarks". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[24] Kameswar Rao Modali. "Assessing the Quality of Zarr Datasets from Km Scale ESM Simulations". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[25] Andrew Kwong. "Enhancing Scientific Reproducibility in CrocoDash for Regional Ocean Modeling". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[26] Paul Jiang. "Characterizing and Correcting Non-Standard Arithmetic in GPUs for Scientific Computing". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[27] Teo Price-Broncucia. "The Fused Multiply-Add and Global Atmospheric Models: An Investigation into a Surprising Correctness Scenario". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.

[28] Anissa Zacharias. "A Short History of Artificial Intelligence and What It Can Teach Us". In: *2nd Workshop on Correctness and Reproducibility for Earth System Software*. Boulder, CO, 2025.