Hacking Km-Scale Models: A Participative Model for Climate Information

Andrew Gettelman[1*], Pier Luigi Vidale[2], Bjorn Stevens[3], Florian Ziemen[4], Zhe Feng[1], Heike Konow[3], Tobias Kölling[3], Lukas Kluft[3], William Jones[5], Sara Pasqualetto[6], Yuting Wu[3], Saskia Brose[7], John Clyne[8], Lluís Fita[9,10], Samuel Green[11,12], Lucas Harris[13], Melissa Anne Hart[11,14], Julia Kukulies[8], Brian Medeiros[8], Timothy M. Merlis[15], Mark Muetzelfeldt[2], Robert Pincus[16], Rosmeri P. da Rocha[17], Masaki Satoh[18,19], Hang Su[20], Daisuke Takasuka[21], Chris Terai[22], Paul Ullrich[22], Tianjun Zhou[20]

1 Pacific Northwest National Laboratory, Richland, USA
2 University of Reading, Reading, UK
3 Max Planck Institute for Meteorology, Hamburg, Germany
4 Deutsches Klima Research Zentrum, Hamburg, Germany
5 Oxford University, Oxford, UK
6 Earth System Modelling and Observations International Project Office, World Climate Research Programme, and Deutsches Klimarechenzentrum GmbH, Hamburg, Germany
7 European Space Agency, Frascati, Italy
8 National Center for Atmospheric Research, Boulder, USA
9 UBA-CONICET, Centro de Investigaciones del Mar y la Atmósfera (CIMA), Argentina
10 CNRS–IRD–CONICET–UBA, Institut Franco-Argentin d'Études sur le Climat et ses Impacts (IRL IFAECI). Argentina.
11 Australian Research Council Centre of Excellence for Weather of the 21st Century
12 University of New South Wales, Sydney, Australia
13 NOAA Geophysical Fluid Dynamics Laboratory, Princeton, USA
14 Institute for Marine and Antarctic Studies, University of Tasmania, Hobart, Australia
15 Princeton University, Princeton, USA
16 Columbia University, New York, USA
17 University of São Paulo, São Paulo, Brazil
18 University of Tokyo, Tokyo, Japan
19 Typhoon Science and Technology Research Center, Yokohama National University
20 Institute for Atmospheric Physics, Chinese Academy of Sciences, Beijing,
21 Tohoku University, Sendai, Japan
22 Lawrence Livermore National Laboratory, Livermore, USA

*Corresponding Author Address: andrew.gettelman@pnnl.gov

Abstract

In May 2025 nearly 700 participants from all around the world coalesced at 10 regional nodes, and a few satellite nodes, to take part in a global hackathon of km-scale (horizontal grid spacing < 10 km) regional and global Earth System Models. Exciting science is emerging from these efforts, ranging across novel model analysis, new ways of integrating with satellite data, and emulation with machine learning. New technologies were trialed that enable the community to work in new and complementary ways to democratize access to global information at a local scale from a set of the world's highest-resolution climate models. The hackathon demonstrated how exascale data can be organized to be accessible to anyone. Fundamentally, the community could apply these techniques and technologies to move towards more participative models for co-production and delivery of diverse sources of climate information for climate scientists and citizens alike.

Significance Statement

This report documents the results of a global hackathon for km-scale atmospheric models held in May 2025. Nearly 700 participants at 10 nodes around the world were able to produce more than a thousand plots from the most advanced global and regional atmospheric models. New tools were used to access and analyze these large data sets. Success was enabled by (1) using data at the right scale (2) the ability to access only needed data and (3) a common open source analysis platform to analyze data without copying it. The community could apply these techniques and technologies widely to move towards more participative models for delivery of climate information.

Introduction

Advancing our understanding of some of the most poorly understood features of the Earth system necessitates Earth System Models (ESMs) capable of representing fine-scale processes and features that are parameterized, poorly represented, or even absent in coarser resolution models. It is hypothesized that reduced reliance on physical parameterizations for features like convection should provide greater certainty in their response to changes in external forcing (Klocke et al., 2025). Global and regional ESMs are pushing the frontier of computing to horizontal scales below 10 km (‘km scale’). For the atmosphere, these scales allow important storm systems (tropical cyclones, meso-scale convective systems, fronts, organized convection) to be represented explicitly. However, a frequently cited shortcoming is that these models are expensive to run and develop, and require access to large computational, storage and analysis facilities that are only available to the world's leading scientific laboratories.

This observation leads to the Borges Paradox, whereby the more relevant information becomes the harder it is to access[1]. The paradox is present in the ‘distributive’ model for climate information provision, in which model output is produced, and researchers access and download data through portals. Over the past few years, a number of scientific (nextGEMS) and technical (Pangeo, Xarray, UXarray, Zarr) projects have established a less centralized way to give users access to data. These methods are based on the idea that through a combination of hierarchical data structures, and lazy computation (see Appendix on Technical Details), it is possible to resolve Borges’s paradox for climate information without the need to download a local copy of the data, but rather access and process data where it resides. Cloud-based approaches ensure universal access and data analysis capabilities independent of local computational resources.

In May 2025, the World Climate Research Programme (WCRP) Global Km-scale Hackathon organized by the WCRP Digital Earth Lighthouse Activity was held at 10+ nodes (physical regional hubs where participants gathered) around the world (Figure 1) to test new methods of climate information delivery. The hackathon coordinated simulations and data provision from the latest generation of research models from leading climate modeling laboratories worldwide. The goals of the hackathon were to:

[1] The "Borges Paradox" refers to the central, absurd paradox of infinity presented in Jorge Luis Borges's short story, "The Library of Babel". The Library contains every possible book, but is so vast that finding a meaningful text is practically impossible, rendering total knowledge useless.

- **Promote Global Scientific Collaboration**: By sharing efficient practices in analysis of km-scale global and regional simulations through collaborative workflows.
- **Promote Global Technical Collaboration**: By bringing scientists of all levels together with technologists we identified bottlenecks in workflows, helped promote the adoption of new access and storage technologies, and encouraged innovation.
- **Expand Long-Term Access**: Looking ahead, the hackathon experimented with methods to broaden access to climate information for interested users worldwide. It was the first step in the creation of a public digital commons as envisioned by projects such as Earth Visualization Engines (EVE; Stevens et al., 2025).

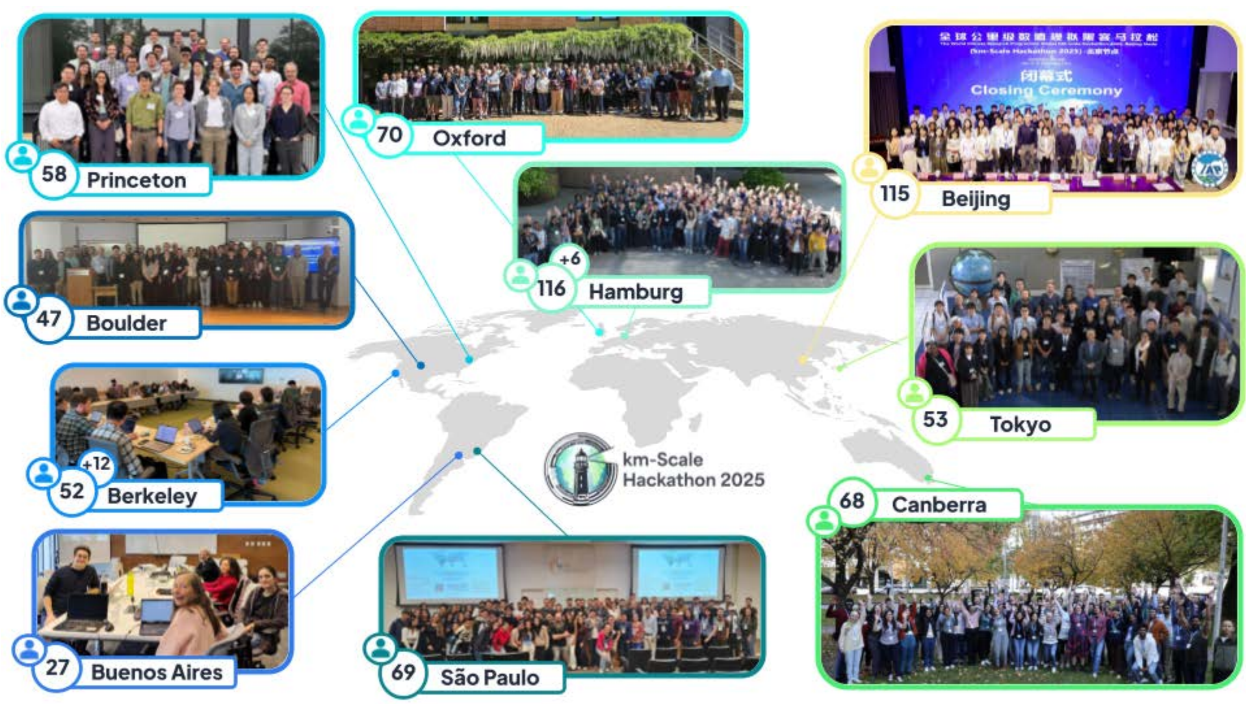


Figure 1: Schematic of Hackathon primary nodes showing location and number of participants

This paper describes the technological innovations of the hackathon that enabled global provision of useful climate information from petabytes of data. We demonstrate how human collaboration was facilitated within each physical node, as well as across nodes spread across five continents (Figure 1) with nearly 700 participants. The hackathon triggered an explosion of coordinated analysis efforts using output from the world's most spatially-detailed global and regional climate models. The profusion of analysis was unprecedented in scale for this sized dataset. The hackathon (and this report) illustrates how emerging technologies of petascale data storage, exascale computing, and new software architectures can enable a more inclusive and participatory delivery of climate information. This report provides an overview of the hackathon, the principles used to build it and the technical and social approaches behind its success, with lessons learned at the end.

Hackathon overview

The goal of the hackathon was to enable the climate science community to analyze global and regional ESMs at km-scale horizontal grid spacing. A new set of model output is now available: for the first time, multiple km-scale global models have run simulations of a year or longer following a coordinated protocol (Takasuka et al., 2024b) with similar (e.g. year 2020) boundary conditions. In addition, regional climate models were run at similar resolutions for even longer (Dominguez et al., 2024; Rasmussen et al., 2023; Senior et al., 2021). Participating models are listed in Table 1.

The scientific foci of the hackathon were framed around questions that could be informed by a year or more of km-scale output from regional and global models. These foci emphasized climate as the statistics of weather, and focused on the two largest forced climate signals, the diurnal and seasonal cycle. Some efforts examined the frequency, distribution, and intensity of precipitation, as well as how precipitation and cloud systems are organized and evolve. Other questions addressed the distribution of precipitation and the effects of fine-scale topography and land-use contrasts (urban versus rural, sea-breeze, etc.). Yet another emphasis was on the representation of the general circulation and the upscaling effect of small-scale phenomena such as precipitation features, convection, and gravity waves on the general circulation (e.g., the seasonal migration of the ITCZ and mid-latitude storm tracks).

Table 1: List of Models

Domain & horizontal grid spacing	Host team	Period	Reference
CAS-ESM			
Global 10 km (partly 5 km)	Beijing	1 year	Zhang et al. (2020)
ARP-GEM			
Global 2.6 km & 1.3 km	Toulouse	1 year	Geoffery and Saint-Martin (2025a,b)
IFS-FESOM			
Global atm 2.8 km / ocean 5 km	Hamburg	2 x 1year	Rackow et al. (2025)
ICON			
Global, 2.5 km	Hamburg	1 year	Hohenegger et al. (2023)
Global + interactive aerosol, 5 km	Oxford	1 year	Weiss et al. (2025)
Unified Model			
Global & regional, 5 km & 10 km	MetOffice	1 year	Willett et al (2025); Bush et al. (2025)
NICAM			
Global, 3.5 km	Tokyo	1 year	Satoh et al. (2014); Takasuka et al. (2024a)
NCAR-MPAS			
Global, 3.75km	Boulder	1 year	Skamarock et al. (2012)
EarthWorks			
Global-Coupled, 15km	Boulder	1-year	Feder et al. (2024)
NCAR-WRF			
Regional: N./S. America, 4 km	Boulder	40 yrs (N.America), 22 yrs (S. America)	Rasmussen et al. (2023), Dominguez et al. (2024)
GFDL-XSHIELD			
Global, 3.25km	Princeton	1 and 2 years	Harris et al. (2023)
DOE-E3SM-SCREAM			
Global, 3.25 km / 128 levels	Berkeley	1 year	Donahue et al. (2024)

These broad questions and others were proposed in a bottom-up fashion by ‘science teams’ self-organized around particular interests (https://digital-earths-global-hackathon.github.io/hk25/scienceteams/, see Gettelman et al. [2025]). Although many teams operated in-person at a single node to benefit from the physical proximity of the scientific members, some teams spanned across nodes. In addition to these activities there were several ‘cross-cutting’ activities that also spanned across nodes: these included groups working on bringing observations to the same formats of the models coordinated around measurements from the new ESA/JAXA Earth Explorer satellite EarthCARE (Illingworth et al., 2015; Wehr et al., 2023); another group using a machine learning emulator for climate models (Brenowitz et al., 2025); several groups focusing on tracking storms (e.g., convective systems, tropical cyclones) in the simulations; and a group that used the hackathon to explore access to pre-operational km-scale climate information being generated as part of the Destination Earth projects.

The physical nodes of the hackathon and their participant numbers are illustrated in Figure 1. There were 10 primary nodes, and three ‘satellite’ nodes in Richland Washington, USA (the Pacific Northwest National Laboratory), Frascati, Italy (ESA-ESRIN, European Space Agency), and Stockholm, Sweden. Satellite nodes used a data center at one of the primary nodes. In addition, there were many remote participants, especially in the Buenos Aires node. In total there were nearly 700 participants. Enabling engagement via regional nodes reduced travel costs, improved accessibility, and preserved the physical contact that underpins human relationships and the development of deeper collaboration. Local nodes were responsible for their participants, simplifying planning.

Principles

The hackathon developed its technical approach following principles that flowed from the participative model and the size and complexity of the model data sets. First, the models mostly, albeit not entirely, followed the protocol of Taskasuka et al. (2024b) who recommended horizontal grid spacing of < 5 km and a year-long simulation. The goal of the hackathon was to enable in-situ data access, on the one hand, and to support methods of data structuring and storage that enable on the fly distribution over low-bandwidth networks. For in-situ access, analysis was performed on high-performance computers where the data was mounted. Individual nodes typically had up to 500Tb of data, some available for remote distribution. For remote distribution, cloud object storage enabled open access to the data. In either local or remote use, lazy computation and hierarchical data storage was key to minimizing the Borges Paradox and making analysis practical at reasonable speed. The key to this new approach is to request data at the right scale, and enable reading the minimal amount of data for the needed visualization or analysis. Common and open source frameworks, platforms and formats were used as much as possible. These efforts were intended to enable in-person and remote collaboration on common science problems, and so required software platforms to foster communication.

Technical Approach

The technical framework underlying the hackathon is described in detail in Ziemen et al. (in prep), but a high-level description is provided here and illustrated in Figure 2. The first component is the data description for model output, and observational products, i.e., reanalyses and observations (mostly from satellite). The model output and observational products were converted from their native grids to a common equal-area grid called HEALPix (see Appendix on Technical Details), all but a few specialized analyses were done on the HEALPix grid. A key feature of HEALPix is the ability to use hierarchical grids with different resolutions. For a global map with a time average (even of just 6 or 24 hours), 3km raw pixels are not needed, and data at much lower resolution, traceable to the highest resolution, can be used. Multi-resolution data stores might seem to compound the data storage problem, but a full HEALPix grid, with successive zoom levels associated with a four-fold finer data representation, requires considerably less storage than a single lat-lon grid at just the highest resolution (63% reduction from a full Gaussian or Lat-Lon Grid due to fewer grids towards the poles). Another feature is a data format that enables accessing variables, regions, and times that are desired, without loading any extra data. For example, for a user wanting to look at data over Europe with one variable, only that variable would be loaded over a small portion (2%) of the planet. A format enabling this (Zarr) is described in the Appendix. As illustrated in Figure 2, data could be local (on a traditional file system) or on a remote cloud storage device (see Appendix). Regardless of where data is stored, this method still enables reading of just the needed data (e.g. just data over Europe), reducing the need to transfer large amounts of data for any analysis.

Figure 2: Schematic of the Hackathon technical stack.

Between the data storage and the user analysis software the hackathon used a common ‘catalog’ to organize the data, visually illustrated in Figure 3. The catalog contains two basic elements: (1) a descriptor of each data set, including metadata and how the data is organized (output frequency, zoom levels, etc.) , and (2) actual locations of the Zarr store paths on a local system. Similar methods have been used by the Pangeo project for traditional GCM output. The catalog also contains descriptors of data stored on object storage in remote locations. The catalog is built for each node’s local data, and then combined with an ‘online’ catalog into a single file that can be accessed by a URL. Thus, the same catalog file is used at every node. The benefit of the catalog approach is that it removes from the user the burden of finding and referencing the different data needed for analysis, unifying data access through a relatively simple interface.

A user simply specifies which node they are at, and the catalog points to where their local data is, and where to find the online data on object storage. Any data within the catalog can be accessed by a reference (typically the model name). Ziemen et al. (in prep) describe the system and further describe a tool that collects all the catalog information onto a web page (displayed in Figure 3), from which the user can see the datasets, descriptors, metadata, and which nodes have the data available locally (blue) and the originator of the data (white). An example of how easy this makes the data access and plotting is illustrated in Figure 4, where four lines of code read and plot surface temperature data from object storage. The catalogues also enabled redundancy. Some data was found to be corrupted at one node, and another node had a 24 hour data center shutdown during the hackathon. In both cases it was possible to fall back on data that was available online from other nodes, and some of the nodes mostly worked with remote data.

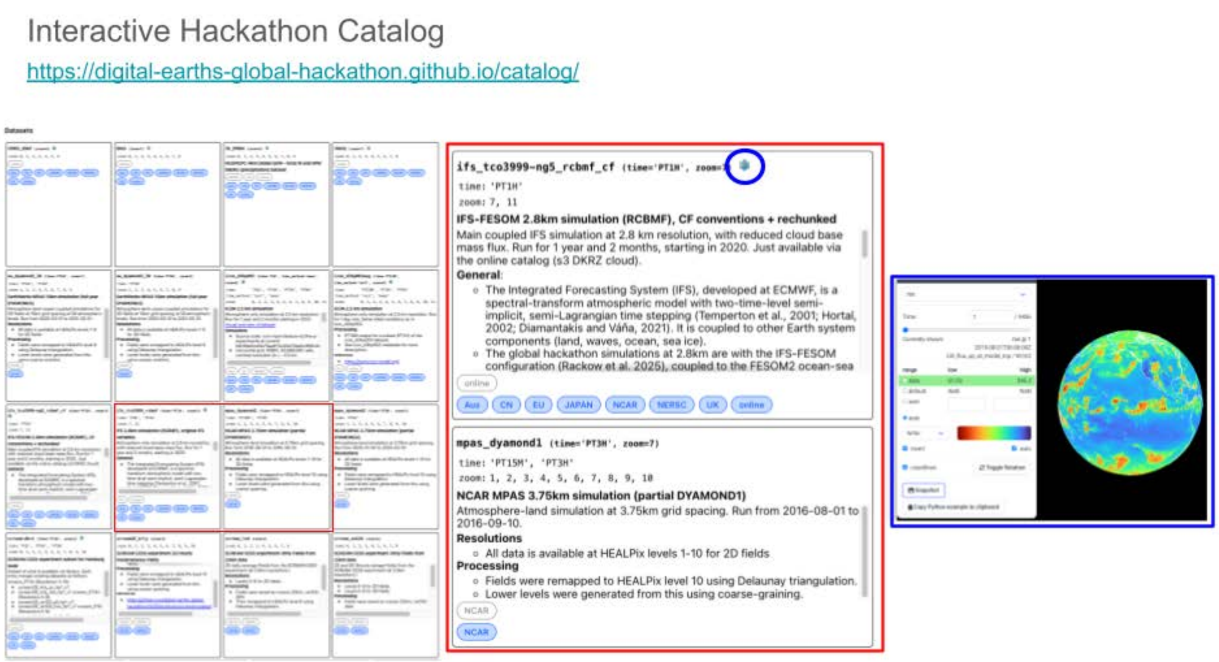
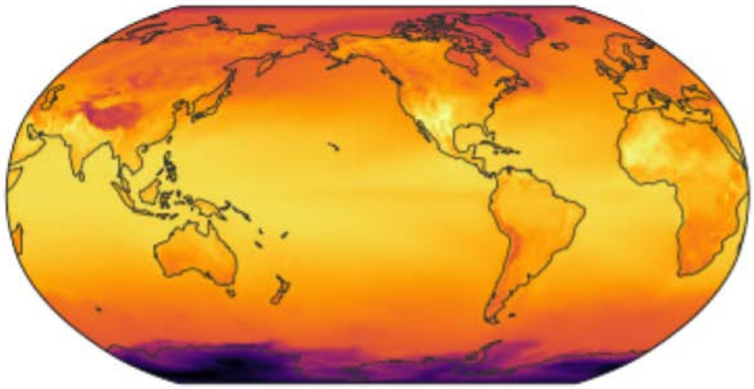


Figure 3: Illustration of the Hackathon catalog. Left: partial entries for observations (top row) and models. Two models in the red box on the right are expanded (center).

Clicking on the icon in blue launches an interactive viewer (right). An interested reader could visit the catalog GitHub site and explore the data.

Finally, there is the user platform for analysis. We developed a common platform by assuming all analysis was to be performed in Python and providing a standard environment of the same set of Python packages. This environment can be described in a list of packages and installed with common tools on systems from laptops to supercomputers. We also used a common Integrated Development Environment (IDE): JupyterLab, but the workflow is compatible with other environments, and many participants used different IDEs (e.g., VSCode). Many larger nodes ran JupyterHub servers and pre-installed the environment, so getting users up and running on the platform required just using a web browser to log into a JupyterHub server. This means that analysis could be done with a slow connection and limited access to computing or storage by just pointing a web browser to a remote data center. This was also successfully demonstrated during the hackathon, when some participants who could not travel were able to participate fully remotely, and others were able to log in to multiple nodes across the planet at the same time for analysis. There were nodes and individuals with lower bandwidth and storage. They were able to do analysis for small regions (a country or metropolitan area) and/or using lower resolutions in the HEALPix hierarchy. The common platform and catalog meant that code developed at one node for one particular model could be shared and run at any node for any other model, or even on a laptop running the Jupyter IDE. Sample notebooks and tutorials were provided for the platform (Figure 4 is from a tutorial). As a result, almost all the participants were able, after an hour or two, to start accessing data and making plots (see Figure 4 caption to try yourself).



```
loc="https://digital-earths-global-hackathon.github.io/catalog/catalog.yaml"
cat = intake.open_catalog(loc)[“online”]
ds = cat["icon_d3hp003"](zoom=7).to_dask()
egh.healpix_show(ds["ts"].sel(time="2020-05-10T00:00:00"), cmap="inferno")
```

Figure 4: Simple plot of instantaneous surface temperature for the ICON model at zoom level 7 (~50km) for one particular time. Generated with the code in the figure. A reader familiar with python environments could simply install the hackathon environment (from https://github.com/digital-earths-global-hackathon/tools/blob/main/python_envs/environment.yaml) and run the notebook for Figure 4 (https://github.com/digital-earths-global-hackathon/hk25-teams/blob/main/hk25-tutorials/simple_plot.ipynb) to generate this plot.

Coordination: Human Approach

The goal of the technical stack was to be able to bring people to the data, wherever it might be, and to deliver just what they need, and no more, to wherever they might be. To facilitate communication, we relied on a loose structure within and across nodes, facilitated by a combination of several different common communication tools.

The organization of the hackathon was distributed. A core group consisting of science, communication, and software leads (~10 people) helped liaise with the technical staff, organize and advertise, recruit scientists, and decide on scope. This group communicated with a larger group of node organizers. Ideally there was a science, logistical, and technical organizer for each node (often these roles were combined in some form, and for smaller nodes they were sometimes one amazing person). The node organizers were the primary unit of communication and organization, though it was very important to have technical and communication staff on the coordinating team with dedicated time to spend on the hackathon.

The technical stack was developed with volunteers (both science and technical staff) to develop test cases of the workflow to prototype reformatting scripts, catalog templates, and plotting examples. Time was key: planning took nearly a year from conception to execution. A Gantt chart of the process is in Figure 5. The technical stack was prototyped 4-6 months ahead, with sample data, and scripts to produce it available 3 months ahead from at least 1-2 models. This was critical in enabling scientists without extensive technical background to help with reformatting the data. Reformatting (HEALPix interpolation and saving in Zarr format) was resource and time intensive and could still improve further. Throughout this process there were regular video conferences (using Zoom) with the various groups (coordination, node organizers, technical group) meeting as needed (more frequently as the time got closer). In addition, an instant messaging server (using open-source Mattermost software) was set up to provide a communications platform for coordination and organization of the nodes. Instant messaging was invaluable for sharing technical knowledge between the nodes before, during, and after the hackathon.

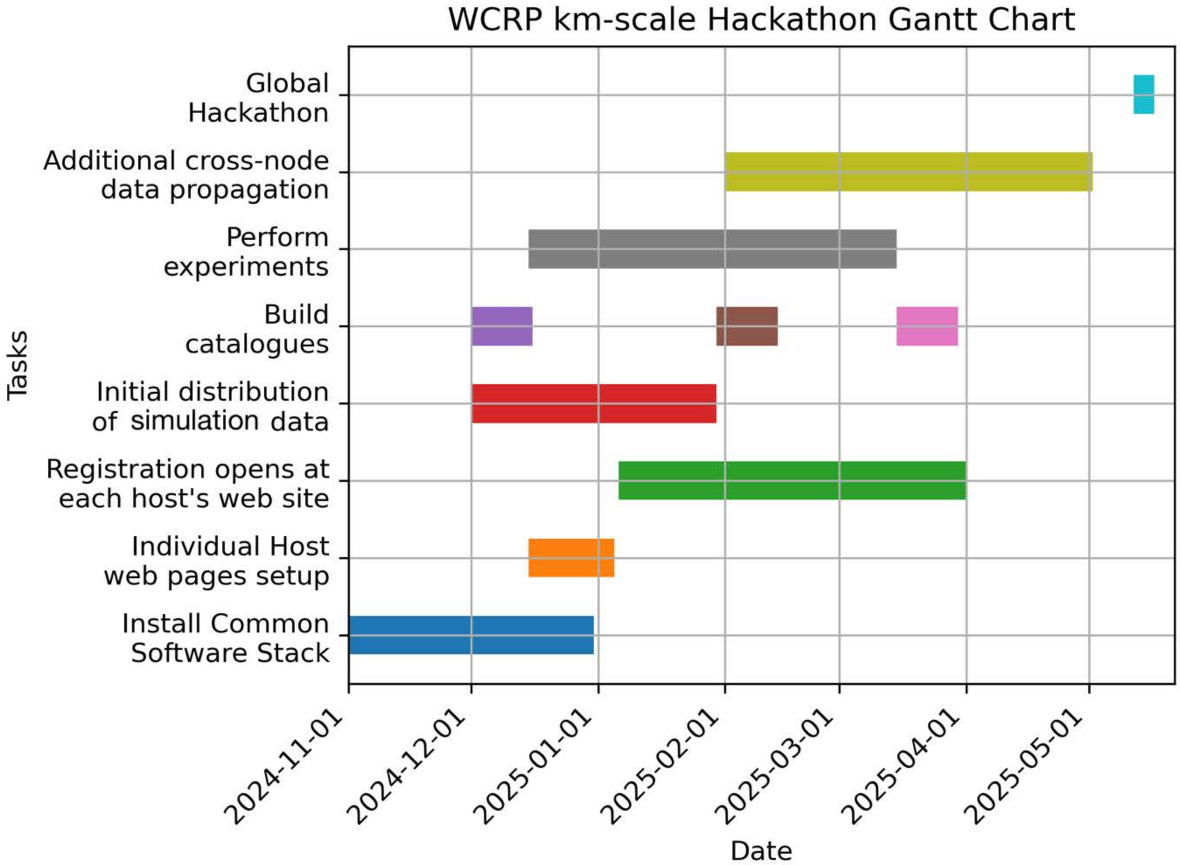


Figure 5: Gantt Chart showing how hackathon tasks were completed over time before the actual event at the top. Catalog tasks were distributed and so noted separately.

Nodes were in full control of their schedules and selection of participants. Some nodes had travel funding for participants, some charged registration fees (depending on their arrangements for rooms and food), and some did not. A very small number of participants (8) were funded directly from Digital WCRP Earth Lighthouse Activity funds.

During the hackathon, multiple methods of communication were layered on top of each other. The basic unit was organizing participants within each node into 'science teams' (https://digital-earths-global-hackathon.github.io/hk25/scienceteams). These were self-organized based on participants' interests and the broad hackathon science topics. Usually they were within a node, but some of the teams crossed nodes. There were also three cross-cutting (and cross-node) activities on (1) observations, (2) feature tracking, and (3) machine learning. The science teams each had their own space on GitHub (https://github.com/digital-earths-global-hackathon/hk25-teams), to have ‘discussions’ and to share code. Nodes also had their own directories to share sample analysis code. Each science team also had access to Mattermost channels for messages within their team, especially when the teams spanned nodes. The channels were very important in facilitating easy communication both within and across nodes.

Many of the nodes had technical staff available in person to troubleshoot with participants and field questions. This was very valuable, for the participants given the range of new technologies being used. Interacting with hackathon participants was also valued by the technical staff, who were hearing users' concerns and technical problems, which could be fixed in real time. For example, the catalog was continuously updated during the hackathon as some groups brought observational data sets that could be shared.

What we learned

The resulting platform enabled individual model analysis efforts across the globe. We estimate that volume of analyses and the number of scientists who have worked with km-scale models has increased tremendously.. As evidence of this, a ‘1000 plot challenge’ was started during the hackathon and reached 1011 plots and subplots that participants shared by the end of the week. A global ‘plot collage’ (Figure 6) illustrates the diversity of what was possible. Some analyses were simple plots of individual fields from models. Other analyses were more complex, and relied on storm tracking in different models (this was coordinated as a cross-cutting team and some of the work was done before the hackathon). Many of these analyses are moving forward to scientific papers, and a mechanism for sharing the results with the model development groups is being designed so that the results can feed back on the model development.

The team concept contributed substantially to the success of the hackathon. It enabled participants to not just learn in a tutorial fashion from the organizers about how to use the technical stack, but to benefit from the diversity of expertise across participants, both within a node and across nodes. Teams were mostly self-organized, and even individuals with projects were grouped into loose teams. Teams were able to ask that their code be run at different nodes to generate plots and data from different models, typically for high-resolution output only available at a single node. The cross-cutting teams were able to train a variety of people to work with complex object tracking codes for the models and with Machine Learning model emulators. Much of the analysis has been archived to GitHub so that it can continue to be developed and others can access it. The whole technical stack has also been archived to GitHub (Gettelman et. al, 2025).

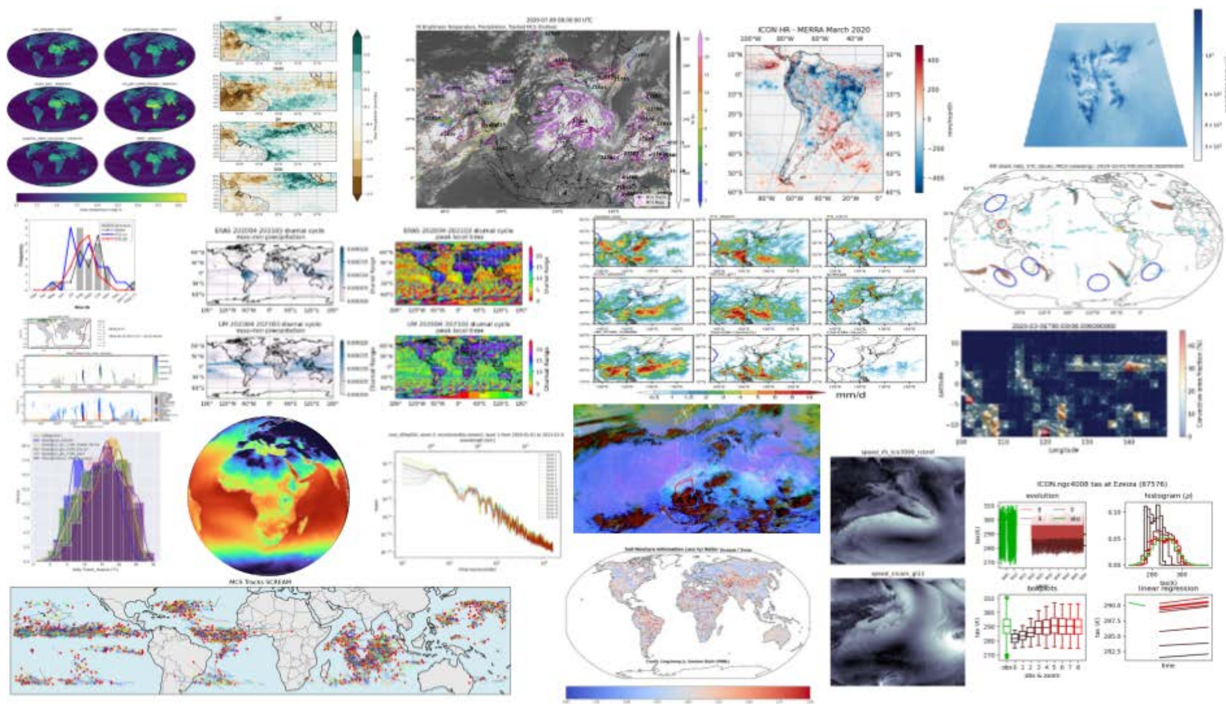


Figure 6: Hackathon Plot Collage with sample plots from each node. All plots were produced from model output during the week of the hackathon. For a more detailed look at the analysis conducted, visit the GitHub science teams site: https://github.com/digital-earths-global-hackathon/hk25-teams

There were several overarching technical achievements as well (detailed in Ziemen et al. (in prep)). The use of catalogs was essential at enabling seamless access of data locally and remotely, largely transparent to the user. Storing the data on the hierarchy of HEALPix resolutions was very effective: the low resolutions were valuable for prototyping analyses or doing work at a broad scale, the high resolutions allowed for detailed analyses of small-scale phenomena. The hackathon also demonstrated that HEALPix, despite being a natural global grid, is able to work well with regional grids. The use of a single set of grids was valuable and streamlined intercomparison as analysis codes could run on any model with minimal effort. Several different software packages (e.g., EasyGEMS, UXarray) for unstructured grids were used, and these each have their advantages and drawbacks, but are being developed rapidly, along with the rest of the software stack. The hackathon also motivated development of software packages before, and during the event.

The use of Python-based Jupyter interactive notebooks and servers was a mature and stable technology that enabled the success of the hackathon. The Python-based platform had many advantages: (1) it allowed the participants to rapidly spin up scientific analysis (bypassing many of the teething problems often encountered), (2) many packages for data analysis and processing addressing user needs are available for unstructured grids in python, (3) it enabled the exact environment to be shared so code developed at one node could be used at any node, (4) by supporting secure server access through a web browser, even low-bandwidth connections for virtual participants could be supported, and (5) Jupyter servers are a secure environment that most computing centers, even those with high security protocols, are able to stand up and provide outside access to. Additionally, most participants had past experience with python.

The hackathon was valuable for identifying key software limitations and gaps, and pathways towards better solutions. While the catalog creation process was straightforward (editing an YAML file), it was done manually and could be improved. The use of Zarr storage was vital for access to the large data sets in an efficient manner, both locally and remotely, but the Zarr portion of the workflow (both creation and access) could also be improved. Implementing a pipeline for Zarr storage and catalogs for the hackathon relied on having technical expertise plus a few individuals at different nodes (mostly scientists) who were willing to take time to understand and implement the pipeline. Some challenges working with data on unstructured grids remain, such as operations related to spatial gradients (e.g., derivatives) and image processing routines that typically require data on a Cartesian grid (e.g., feature identification and tracking). Some impact models (e.g. hydrology) require mass-conservative re-projection to traditional grids. Investments on improving software support of unstructured grids would greatly benefit the scientific community in adapting to the km-scale climate data on the HEALPix (or any unstructured grid) framework. Most analysis desired by users during the hackathon could be done on a common (and non-conservative) grid, with only a few detailed analyses (e.g. derivatives, budgets), needing the native grid.

Summary

The km-scale model hackathon demonstrated that we can move from a distributive to a participative model for climate information, and how this can include more people from more locations, ranging from some of the largest computing centers in the world to individual universities or even remote connections on laptops in the Global South. The distributive model fosters collaboration and broadening access to the best high fidelity climate information from regional or global simulations. We expect peer-reviewed journal articles to emerge shortly from the analysis of the one-year long simulations, including the first multi-model global km-scale intercomparison for today's climate.

The hackathon was enabled by a combination of technical innovations to enable appropriate use of data at the right scale, and to serve up just the data that was needed for participants. We were able to use common analysis platforms (Python) to create a common global interface for the data that was easy to implement at 10+ nodes on heterogeneous infrastructure all over the planet. This common interface was critical from a user perspective to reduce the learning curve for working with the data. The technical stack was developed through a close iteration of science and technical planning. Some key innovations included the use of hierarchical grids and cloud native storage architectures, as well as the use of catalogs to describe, organize, and access the data. The result was a common set of standards to get disparate models (including regional and global models) into the same grids. Native model grids were not necessary for most of the analysis users performed. Distributed organization allowed for each node to tailor their experience for local conditions and resources, as well as to share and distribute common tasks. Availability and engagement with technical staff during the hackathon was critical for rapid problem solving. The hackathon also revealed some limitations with current tools, and several avenues for evolving and improving the technical stack for future use are now clear: use of better storage formats (Zarr v3), more robust or automated catalogs, and the need for continued development of tools working on non-Cartesian unstructured grids.

The long-term vision is to develop robust and simple tools and methods to enable broader and more effective application of climate information. The tools and technologies exist, though need to be improved. Key next steps are expanding use of the cloud storage and common platform idea to operationalize it, along with continued education of users with these new methods. We are already working to advance the technical stack as noted above. Furthermore, we are working with the larger global and regional climate modeling community through the WCRP to operationalize these methods.

The hackathon demonstrated that appropriate technology actually can be used to make the most advanced climate information more accessible and inclusive. The approach could also allow us to bring together different sources of information: not just global and regional models, but also Machine Learning models and observations. We now are at the stage where we can think about expanding such tools beyond climate scientists to a broader community desiring climate information. Over 700 users were able to use the world's largest climate models and in a week created over 1000 plots and analyses. Who knows what they will do next?

Acknowledgements

The Pacific Northwest National Laboratory is operated for the U.S. Department of Energy by the Battelle Memorial Institute under contract DE-AC05-76RL01830.

Lawrence Livermore National Laboratory operates under the auspices of the U.S. Department of Energy under contract DE-AC52-07NA27344. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a Department of Energy User Facility (projects m4581 and m1867). Diagnostics in Buenos Aires node were performed using CIMA-FAECI's Papa-Deimos, Argentina, partially funded by INSU project LEFE ref. 12962. This work was supported by the University of Reading Advancing the Frontiers of Earth System Prediction (AFESP) Programme [award number A3720200]. This work was also supported by the National Center for Atmospheric Research (NCAR), which is a major facility sponsored by the NSF under Cooperative Agreement No. 1852977. The South America participants were supported by Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP: 2025/01342-0) and Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq: 440363/2025-0). Participant travel support to Princeton and NCAR was provided by NSF [award number 2524887]. This work was supported by the nextGEMS project under the European Union's Horizon 2020 research program (Grant No. 101003470) and used resources of the Deutsches Klimarechenzentrum (DKRZ) granted by its Scientific Steering Committee (WLA) under project IDs bb1153 and bm1235. The Australian node was supported by the Australian Research Council Centre of Excellence for the Weather of the 21st Century (CE230100012) and the Australian Government's National Collaborative Research Infrastructure Strategy (NCRIS), with access to computational resources and support provided by the National Computational Infrastructure (NCI) and ACCESS-NRI. The Tokyo node was supported by JSPS Core-to-Core Program, "International Core-to-Core Project on Global Storm Resolving Analysis" (Grant Number: JPJSCCA20220001).

Appendix: Hackathon Technical Details

The Hierarchical Equal Area isoLatitude Pixelation (HEALPix) hierarchical grid (Górski et al., 2005) was originally developed to look at cosmic microwave background whole sky imagers on a spherical grid. A key feature of HEALPix is that the structure is hierarchical: the grids are defined by different zoom levels, starting with 12 quadrilateral cells tessellating a sphere, successively refined by subdivision into four new cells, thus providing a hierarchy of “zoom” levels. This enables use of data at an appropriate scale for a particular analysis operation. For a zonal mean estimate, for example, lower zoom levels can be used. Displaying a global image at typical size in a full HD presentation (1 million pixel) resolution would only require zoom level 8, approximately 25 km; zoom level 10 is about 6 km and has 12M cells. There is no need to read in 12M cells to plot 1M pixels. This modest decrease of two resolution steps already yields a data reduction by 94%.

Data storage format was critical for enabling distributed access. Hackathon participants across 10 global nodes needed to access petabyte-scale climate model outputs without full downloads. Zarr met this requirement by loading dataset structure from a single metadata file, then transferring only the requested spatial or temporal blocks. This avoided the overhead of transferring thousands of individual NetCDF files over network connections—the standard output format from model runs. We thus repackaged model outputs into Zarr stores, using the Zarr v2 format for compatibility with existing analysis tools.

Zarr stores are directories with a name ending in .zarr. Within the top-level directory are subdirectories containing small blocks (aka “chunks”) of data with a typical size of 8-20 megabytes. A metadata file describes where variables and their space and time dimensions are located within the directory tree of chunks. This organization facilitates very fast reads of individual parts (chunks) of the data, and for reading large sections of data, it enables it to be done in parallel. However, Zarr v2 creates thousands to billions of small files (one per chunk), which most filesystems are not designed for. This posed challenges in creating, moving, and storing Zarr collections. For users reading data during the hackathon there was minimal impact. Fortunately, the Zarr v3 standard (introduced after the hackathon software stack was developed) has a solution to this problem by grouping chunks into a single file (or shard), so instead of 10,000 files for a 100 GB data set with 10 MB chunks, there could be 100 chunks in a shard and only 100 files. Zarr also supports fast compression (which was used in the hackathon) and is designed to work with conventional file systems and object storage. The Zarr API does not provide support for hierarchical data. Thus, to accommodate HEALPix multiple zoom levels were created as separate Zarr stores (Figure 2, colored cylinders).

Complementing our choice of Zarr, was our use of Xarray based Python tools. Xarray offers lazy access: data is only loaded when it is needed. Xarray has become a community standard for analyzing large geophysical data sets. Together with Zarr and HEALPix, working with remotely stored exascale data volumes was as easy as using traditional analysis methods on structured and locally stored laptop-scale data volumes.

Several centers were able to take full advantage of these new workflows to make the data openly available across nodes. Object storage is a data storage architecture that organizes information as individual units called "objects", which consist of the data and metadata, and a unique identifier, defined by an interface accessible via standard web protocols. Object storage is a core component of most cloud storage systems. For the hackathon, several centers provided data using the https protocol to provide access to Zarr stores of output. Zarr and object storage enables reading only the needed ‘chunks’ over the internet, as if from disk. Some of the data was copied to different nodes, with larger nodes able to host multiple datasets, and offer data-proximate computing for higher resolution zoom levels.

References

Brenowitz, N. , and Coauthors, 2025: Climate in a bottle: Towards a generative foundation model for the kilometer-scale global atmosphere. *arXiv preprint arXiv:2505.06474*

Bush, Mike, and Coauthors 2025. “The Third Met Office Unified Model–JULES Regional Atmosphere and Land Configuration, RAL3.” *Geoscientific Model Development* **18** (12): 3819–55. https://doi.org/10.5194/gmd-18-3819-2025.

Dominguez, F., and Coauthors, 2023: Advancing South American Water and Climate Science Through Multi-Decadal Convection-Permitting Modeling, *Bull. Amer. Meteor. Soc.*, **104**, E32-E44, https://doi.org/10.1175/BAMS-D-22-0226.1

Donahue, A. S., and Coauthors 2024. “To Exascale and Beyond—The Simple Cloud-Resolving E3SM Atmosphere Model (SCREAM), a Performance Portable Global Atmosphere Model for

Cloud-Resolving Scales.” *Journal of Advances in Modeling Earth Systems* **16** (7): e2024MS004314. https://doi.org/10.1029/2024MS004314.

Feder, A., D. Randall, and D. Dazlich. 2024: Effects of Warming and Stratospheric Aerosol Injection on Tropical Cyclone Distribution and Frequency in a High-Resolution Global Circulation Model. *EGUsphere*: 1–35. https://doi.org/10.5194/egusphere-2024-3526.

Geoffroy, Olivier, and David Saint-Martin. 2025a. “The ARP-GEM1 Global Atmosphere Model: Description, Speedup Analysis, and Multiscale Evaluation up to 6 Km.” *Journal of Climate* **38** (18): 4739–62. https://doi.org/10.1175/JCLI-D-24-0547.1.

Geoffroy, Olivier, and David Saint-Martin. 2025b. “Global Kilometer-Scale Simulations with ARP-GEM2: Effect of Parameterized Convection and Calibration.” arXiv:2511.00829. *arXiv*, November 2. https://doi.org/10.48550/arXiv.2511.00829.

Gettelman, A., Y. Wu. P. L. Vidale and B. Stevens, 2025: digital-earths-global-hackathon/km-scale-hackathon-2025: v1.0 (v1.0-release). Zenodo. https://doi.org/10.5281/zenodo.15650915

Górski, K. M., and Coauthors 2005: HEALPix: A Framework for High-Resolution Discretization and Fast Analysis of Data Distributed on the Sphere. *The Astrophysical Journal* **622** (2): 759. https://doi.org/10.1086/427976.

Harris, Lucas, Linjiong Zhou, Alex Kaltenbaugh, Spencer Clark, Kai-Yuan Cheng, and Chris Bretherton. 2023. “A Global Survey of Rotating Convective Updrafts in the GFDL X-SHiELD 2021 Global Storm Resolving Model.” *Journal of Geophysical Research: Atmospheres* **128** (10): e2022JD037823. https://doi.org/10.1029/2022JD037823.

Hohenegger, Cathy, and Coauthors 2023. “ICON-Sapphire: Simulating the Components of the Earth System and Their Interactions at Kilometer and Subkilometer Scales.” *Geoscientific Model Development* **16** (2): 779–811. https://doi.org/10.5194/gmd-16-779-2023.

Illingworth, A., and Coauthors, 2015: The EARTHCARE satellite: The next step forward in global measurements of clouds, aerosols, precipitation and radiation. *Bull. Amer. Meteorol. Soc.,* **96**, 1311-1332. http://dx.doi.org/10.1175/BAMS-D-12-00227.1

Rackow, Thomas, and Coauthors 2025. “Multi-Year Simulations at Kilometre Scale with the Integrated Forecasting System Coupled to FESOM2.5 and NEMOv3.4.” *Geoscientific Model Development* **18** (1): 33–69. https://doi.org/10.5194/gmd-18-33-2025.

Rasmussen, R. M., and Coauthors, 2023: CONUS404: The NCAR–USGS 4-Km Long-Term Regional Hydroclimate Reanalysis over the CONUS. *Bull. Amer. Meteorol. Soc* **104**, E1382-E1408. https://doi.org/10.1175/BAMS-D-21-0326.1.

Satoh, M., and Coauthors, 2014: The Non-hydrostatic Icosahedral Atmospheric Model: Description and development. *Progress in Earth and Planetary Science*. **1**, 18. https://doi.org/1186/s40645-014-0018-1

Senior, C. A., and Coauthors, 2021: Convection-Permitting Regional Climate Change Simulations for Understanding Future Climate and Informing Decision-Making in Africa. *Bull. Amer. Meteor. Soc.*, **102**, E1206–E1223, https://doi.org/10.1175/BAMS-D-20-0020.1

Skamarock, William C., Joseph B. Klemp, Michael G. Duda, Laura D. Fowler, Sang-Hun Park, and Todd D. Ringler. 2012. "A Multiscale Nonhydrostatic Atmospheric Model Using Centroidal Voronoi Tesselations and C-Grid Staggering." *Monthly Weather Review* **140** (9): 3090–105. https://doi.org/10.1175/MWR-D-11-00215.1.

Stevens, B., and Coauthors, 2024: Earth Virtualization Engines (EVE). *Earth Syst. Sci. Data*, **16**, 2113–2122. https://doi.org/10.5194/essd-16-2113-2024

Takasuka, D., and Coauthors: 2024a: How can we improve the seamless representation of climatological statistics and weather toward reliable global K-scale climate simulations? *Journal of Advances in Modeling Earth Systems*, **16**, e2023MS003701. https://doi.org/10.1029/2023MS003701

Takasuka, D., and Coauthors: 2024b: A Protocol and Analysis of Year-Long Simulations of Global Storm-Resolving Models and Beyond. *Progress in Earth and Planetary Science* **11** (1): 66. https://doi.org/10.1186/s40645-024-00668-1.

Weiss, P., Herbert, R., & Stier, P. 2025: ICON-HAM-lite 1.0: Simulating the Earth system with interactive aerosols at kilometer scale. *Geoscientific Model Development*, **18**(12), 3877–3894. https://doi.org/10.5194/gmd-18-3877-2025

Wehr, T., and Coauthors, 2023: The EarthCARE mission – science and system overview, *Atmos. Meas. Tech.*, **16**, 3581–3608, https://doi.org/10.5194/amt-16-3581-2023.

Willett, Martin Richard, and Coauthors 2025. "The Met Office Unified Model Global Atmosphere 8.0 and JULES Global Land 9.0 Configurations." *EGUsphere*, May 28, 1–55. https://doi.org/10.5194/egusphere-2025-1829.

Zhang, He, and Coauthors 2020. "Description and Climate Simulation Performance of CAS-ESM Version 2." *Journal of Advances in Modeling Earth Systems* **12** (12): e2020MS002210. https://doi.org/10.1029/2020MS002210.