Expanding Access to Science Participation: A FAIR Framework for Petascale Data Visualization and Analytics

Aashish Panta, Alper Sahistan, Xuan Huang, Amy A. Gooch, Giorgio Scorzelli, Hector Torres, Patrice Klein, Gustavo A. Ovando-Montejo, Peter Lindstrom, Valerio Pascucci

***Abstract*—The massive data generated by scientists daily serve as both a major catalyst for new discoveries and innovations, as well as a significant roadblock that restricts access to the data. Our paper introduces a new approach to removing big data barriers and democratizing access to petascale data for the broader scientific community. Our novel *data fabric abstraction* layer allows user-friendly querying of scientific information while hiding the complexities of dealing with file systems or cloud services. We enable FAIR (Findable, Accessible, Interoperable, and Reusable) access to datasets such as NASA's petascale climate datasets. Our paper presents an approach to managing, visualizing, and analyzing petabytes of data within a browser on equipment ranging from the top NASA supercomputer to commodity hardware like a laptop. Our novel data fabric abstraction utilizes state-of-the art progressive compression algorithms and machine-learning insights to power scalable visualization dashboards for petascale data. The result provides users with the ability to identify extreme events or trends dynamically, expanding access to scientific data and further enabling discoveries. We validate our approach by improving the ability of climate scientists to visually explore their data via three fully interactive dashboards. We further validate our approach by deploying the dashboards and simplified training materials in the classroom at a minority-serving institution. These dashboards, released in simplified form to the general public, contribute significantly to a broader push to democratize the access and use of climate data.**

***Index Terms*—Large Scale Data Management, Data Visualization, Petabytes, Climate Data, Web-based Visualization, Machine Learning Insights, Dashboards, Volume Rendering.**

I. Introduction

WITH the continuous generation of petabyte-scale datasets on a daily basis, effective analysis and visualization are essential to deriving meaningful insights. Leading scientific institutions, such as NASA, play a pivotal role in producing and managing these vast data resources. Although significant resources are poured into making it accessible (e.g. NASA's DYAMOND and ECCO data [1], [2]), inherent challenges persist including: the difficulty accessing the data, the limitations in computational power, and the need for real-time processing capabilities. Downloading petascale data locally is problematic due to limitations of local memory or disk size and insufficient bandwidth for remote disks [3]. Our work focuses on collaboration with these institutions to improve data accessibility, empowering researchers to unlock newer insights within these vast datasets.

Researchers and scientists often want to ask conceptually simple questions on time-series data and slices of volumes. Scientific data visualization on petascale datasets can require up to hundreds of GPU and CPU core hours, requiring hours of waiting in a queue at a busy center. Typically, static visualizations are generated for a selected time range, scalar, region, and resolution, limiting the ability to interactively view the data. Researchers may also need to interactively analyze large datasets, which is difficult with traditional static visualization methods that limit the ability to ask real-time questions and perform on-the-fly analysis. Other technical challenges for big data in domains like climate science include migrating code, analytic products, and large repositories within the growing network of storage and computational resources [4].

Our collaboration with domain scientists and visualization experts has helped to create novel interactive visualization dashboards for petabytes of data with progressive loading and decoupling of the storage infrastructure in order to increase data democratization. Our specific contributions include the following:

- A novel *data fabric abstraction layer* that allows users to request information based on their scientific needs without dealing with the low-level file format specification or network limitations. To mitigate the increasing complexity and volume of data, our data fabric abstraction is built with FAIR (Findable, Accessible, Interoperable, Reusable) principles in mind. Our *FAIR Digital Objects (FDOs)* layer responds to user requests within the specified quality / resource bounds or notifies that the request may need to be revised (e.g., reducing quality or increasing resources).
- Efficient Data Reorganization, Conversion, Reduction/Optimization pipeline that allows efficient storage and data transfer by utilizing compression strategies for data. Transforming data into an *Analysis-Ready Cloud-Optimized (ARCO)* friendly format significantly reduces computational load and storage requirements.
- Scalable Visualization Dashboards enable progressive visualizations of petascale data with advanced analytical tools and a user-friendly design, encouraging scientific

Aashish Panta, Alper Sahistan, Giorgio Scorzelli, Amy A. Gooch, Valerio Pascucci are with the University of Utah, Salt Lake City, UT 84112 USA. E-mail: {aashishpanta0, scrgiorgio, amy.a.gooch, pascucci.valerio}@gmail.com

Hector Torres and Patrice Klein are with NASA Jet Propulsion Lab, Pasadena, California.

Xuan Huang is with 3D Graphics team at Cesium, Philadelphia, PA.

Gustavo A. Ovando-Montejo is with Utah State University, Blanding, Utah.

Peter Lindstrom is with Center for Applied Scientific Computing at Lawrence Livermore National Lab(LLNL), Livermore, CA 94550.

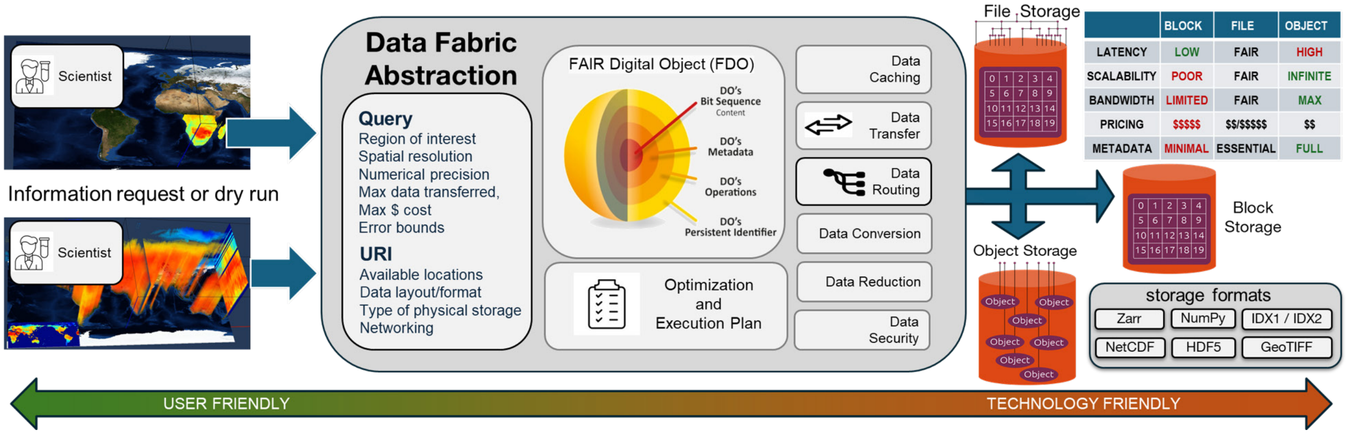


Fig. 1: Our novel data abstraction framework allows a scientist to express a query for the information needed with additional parameters, such as the quality required to achieve a trustworthy result and/or the maximum amount of resources available for its execution. Our Data Fabric Abstraction (middle) handles the query and builds the *Uniform Resource Identifier (URI)*. The *FAIR Digital Object (FDO)* provides the information necessary for an implementation that optimizes the execution of the query (middle). The low-level execution (right) will use the available networking and storage resources, including different file formats and storage models (file systems, object storage, or block storage), as needed.

curiosity and discovery.

- Integration of Machine Learning-Powered Insights into specific dashboards for real-time anomaly detection and interactive analysis of climate data. We leverage a ConvLSTM2D [5] model to dynamically identify spatiotemporal anomalies in climate simulations, enabling users to uncover trends, extreme events, and deviations for more informed scientific exploration.
- Data Democratization via publicly accessible web links to over a petabyte of cloud-optimized data, enhancing accessibility and collaboration. Our approach allows undergraduate students in a minority-serving institution to use in their exercises the same petascale dataset as NASA scientists on their largest supercomputer, seeking to advance access to scientific data and discoveries.

We demonstrate our approach through dashboards available on supercomputers and servers, both accessing publicly available data from cloud storage. We evaluate our dashboards through four use cases: 1) petascale visualization of multiple variables with 1.1 PB data from the cloud; 2) studying relationships between oceanic and atmospheric variables, such as how sea surface temperature affects the formation of ice, water, and clouds in the atmosphere, using cloud-served data; 3) machine-learning powered insights on a 38 TB daily climate simulation spanning 150 years; and 4) demonstrating how enabling FAIR data opens the doors for underserved communities to access data previously out of reach. We examine performance, discuss lessons learned, and review the intellectual merit and societal benefits of our novel approach for petascale data visualization and analysis.

II. Related Works

Visualizing large-scale data directly from a web environment provides unprecedented access to information. The ability to process and render complex datasets from a web browser offers unique advantages in efficient analysis, accessibility, and data management. The shift toward browser-based visualization tools enables users across various disciplines to access and interact with information in real-time without the need for specialized hardware or extensive software installations. As interest and demand for web-based visualization grows, researchers and developers have developed a variety of frameworks, libraries, and methodologies to address the inherent challenges associated with rendering large-scale datasets in a browser environment.

A. Large-Scale Web-based Visualizations

The most popular web-based visualization libraries today include D3 [6], popular for its ability to directly manipulate and transform the content within its *document object model (DOM)*. Several other libraries leverage WebGL to handle large-scale data visualizations efficiently and with high performance directly within a browser. WebGL enables these libraries to provide rich, interactive, and 3D visualizations using the GPU for graphics rendering. Among these are libraries Deck.gl [7], Luma.gl and Three.js [8].

Visualizing large-scale datasets from a browser has been previously researched. Usher et al. [9], [10] developed an isosurface computation algorithm for block-compressed data to visualize a terabyte of scientific data from a browser. We have far exceeded this by creating a framework that can visualize more than a petabyte of data from the cloud. Alder et al. [11] developed USGS National Climate Change Viewer to visualize 17 terabytes of climate projection data from compressed NetCDF-4 files and preprocessed the data for statistics instead of computing them on the fly. Walker et al. [12] worked on 50 MB of geospatial data at once, mentioning a browser limitation that caused excessive latency. Other tools, such as ParaViewWeb [13], perform data processing and rendering on the server side and stream the results back to the client. Mohammad et al. [14] deployed efficient and affordable scientific visualization as a microservice, but noted challenges with inconvenient network latency and costly egress costs. However, none of these systems have been capable of working with petascale datasets.

A framework developed by Lu et al. [15] allows on-the-fly visualization of the multiscale climate datasets but remotely uses cloud services to provide big data processing ability. Ravindu [16] describes loading data into memory as one of the key challenges of visualizing data from a browser. To tackle these challenges, tools such as Firefly [17] use progressive rendering techniques on a 250GB dataset to visualize from a browser. Other challenges mentioned by Khadija et al. [18] include scalability and high-performance requirements for big data analysis and visualization.

B. Climate-specific Data Visualization Tool

Some existing tools, such as Ultrascale Visualization Climate Data Analysis Tools (UV-CDAT) [19], support data gridding and exploratory data analysis, but require users to download sophisticated packages. A Python-based tool called CCPviz developed by Aizenman et al. [20] provides a data processing and visualization module for climate data. However, the CCPviz architecture is inefficient for petascale data as it requires transferring all selected data between the different layers of their architectures. Sun et al. [21] developed a web-based visualization framework for climate data using Google Earth by precomputing all images on the local server and mentioned that on-the-fly data access from remote servers is slow and impractical.

A Web analysis platform called ClimateCharts.net developed by Zepner et al. [22] focuses on the general interactive features of the data, but had issues due to the lack of computational resources and network latency. Another framework developed by Wong et al. [23] enabled interactive visualization of large-scale scalar and flow field data but required significant data downsampling to make their workflow run on the desktop computer. Other challenges mentioned by Wong et al. [24] include being dependent on in-situ data analysis, lack of interaction, visualization techniques, and limited community involvement.

Several implementations endeavor to make the NASA Climate data easier to access [25]–[28]. Such methods require users to access libraries designed to make the data accessible. Still, the ability to slice vertically or horizontally interactively requires manual installation of complex libraries or additional expertise. Scientists and developers have created libraries like xmitgcm [29], [30] to handle these issues for petascale NASA datasets, but they still lack the interactive features that many climate scientists desire. Other tools developed at NASA, such as Podaac [31], provide *near real-time (NRT)* access to some data products but do not give users the flexibility to change the range, colormap, and other utilities such as slicing the data, helpful in navigating the depths of the data. Ellsworth et al. [32] developed a visualization environment for petabytes of data using the hyperwall, a display wall with 128 displays, and associated computing clusters. Whereas high-resolution displays and custom software allowed one to quickly view large amounts of data, this unique system was restricted to scientists who could secure an invitation to the facility.

C. Data Reorganization using OpenVisus

Many scientists need help dealing with massive datasets due to hardware limitations, slow data movement, and I/O bottlenecks. Among other technologies, we use OpenVisus [33], [34], an open source out-of-core data management framework designed to address these issues through data reorganization and multi-resolution access. OpenVisus uses hierarchical Z-order space filling curves to encode spatially coherent access patterns in storage, optimizing both spatial locality and disk access patterns [35]. Data are stored in IDX format, which partitions data into multiresolution blocks organized by resolution and precision levels. This structure enables efficient subsetting, downsampling, and progressive streaming, allowing scientists to interactively query only the data needed for a specific task. The IDX format and OpenVisus API, recognized for its fast and progressive data streaming functionalities, support fast I/O of petascale simulations [36], [37] as well as post-hoc querying and visualizing petascale data in various scientific applications [33], [38], [39]. Designed to provide progressive access for very large datasets, this framework optimally exploits the existing caching hardware in modern architectures. The cache-oblivious approach [40]–[42] exploits this structure by storing large data arrays in a cache-optimized manner. A critical aspect that will be specialized and optimized, especially for the use cases presented in this paper, is the ability to progressively encode the spatial resolution and numerical precision of the data [43], [44], thus minimizing the cost of data movements for any data analysis and visualization workflow [45], [46].

We tackle the challenge of working with datasets that are too large for a system's memory by utilizing OpenVisus' out-of-core computations. We then transform the data into an Analysis-Ready Cloud-Optimized (ARCO) [30], [47] friendly format, which means the data is clean and ready for analysis while also allowing efficient and direct access to data subsets in the cloud. By leveraging the amount of available computational resources a user has, our approach smartly transfers data between the system's fast but limited internal memory and external storage options, which offer more space but are slower. As a result, our framework handles very large datasets that exceed the system's memory limits without significantly slowing down or reducing the effectiveness of data processing.

III. A Novel Data Fabric Abstraction

We introduce a data abstraction layer that concurrently addresses the user's need to access information easily while being able to control the amount of resources used. Through the use cases presented in this paper, we demonstrate how the data abstraction layer aids in visualizing, analyzing, and sharing petascale climate simulation datasets. We show how our framework facilitates the use of these massive datasets for world-renowned climate scientists using NASA's largest supercomputer and for undergraduate students in a minority-serving institution.

Our data fabric abstraction consists of several modules as shown in Fig. 1 including query, universal resource identifier (URI), FAIR Digital Object (FDO), and plan modules, as well as pipelines for data caching, transfer, routing, conversion, reduction, and security. The data fabric abstraction then relies on an API to access file, block, or object storage. The storage formats include ZARR, NumPy, NetCDF, HDF5, GeoTIFF, and IDX1/IDX2. By building this modular abstraction, we

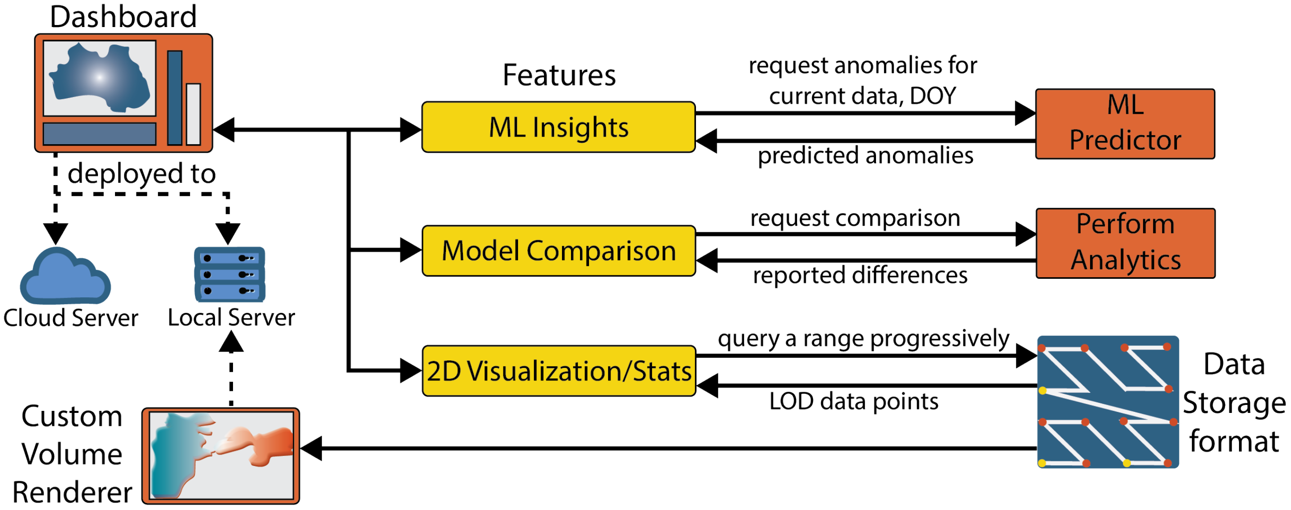


Fig. 2: Overview of the integrated framework for climate data analysis and visualization: Our system connects a Dashboard deployed on either cloud or local servers, providing multi-faceted functionality for climate data exploration. Our framework includes 2D Visualization and Statistical Analysis, Model Comparison, and Machine Learning (ML) Insights for anomaly detection and prediction. Users can interactively query and progressively load data stored in optimized formats, run 2D or 3D visualization, perform general analytics, compare model outputs, and load ML models as required. Additionally, we create a Custom Volume Renderer to enhance the volumetric nature of datasets with addition of highlights, transparency, and shadows.

can easily add, change, and update any of these black boxes through a simple API without worrying about cascading issues.

A. *Query and URI*

Removing the need to manage low-level storage intricacies, we provide an API to request information at a high level of abstraction and include a variety of technical requirements. Typical primary query elements include spatial extent, time value or range, and variables of interest (temperature, salinity, velocity, etc.). Queries are specified not with respect to how the data is stored in a particular file format but in the analysis coordinate system, similarly to an Xarray API [48]. Although Xarray is user-friendly, Xarray does not address the problem of mapping a request to an impractically large amount of resources. Therefore, we introduce additional parameters that the scientific community has identified as necessary when dealing with massive data [49]–[51]. For example, for a given query, the user can specify the spatial resolution and/or numerical precision needed to satisfy the scientific needs. Additional constraints include the maximum cost in egress fees budgeted for data stored in the commercial cloud or the maximum delay between query and response. Since it may not always be possible to satisfy all the requirements, a query may not return the information requested but indicates that the conditions need to be relaxed (Fig. 1, middle).

As users make specific requests within the query abstraction framework, the back-end abstraction *uniform resource identifier (URI)* handles query requests, such as data caching, transfer, routing, conversion, reduction, and security. Each of these is critical for maintaining performance and data integrity across different storage types and formats, whether in block storage, file storage, or object cloud storage. Internally, queries are resolved through a multiresolution data indexing structure that uses hierarchical space-filling curves. Based on query parameters, only the necessary blocks are accessed and streamed progressively: first coarse resolution, then finer, based on the display, analysis, or bandwidth requirements. Flexible back-end logic dynamically selects which resolution or format to serve depending on query cost, access policy, and system bandwidth.

B. *FAIR Digital Object*

To address the complexities of large data distributed across various platforms and the potential use of different file formats, we introduce the first advanced *FAIR Digital Object (FDOs)* framework [52] implementation, which democratizes access to data while following the *FAIR (Findable, Accessible, Interoperable, Reusable)* guiding principle [53] [54], illustrated in Fig. 1. FDOs encapsulate and manage digital content, such as data, models, or workflows. In general, FDO consists of the following layers [55]: digital object (data, code, research outputs), identifiers (unique and persistent such as *Digital Object Identifier (DOI)*, standards and code (typically open file formats), and metadata (provenance). Our advanced FDO includes all the actionable information needed to determine if and how a given query can be resolved (Fig. 1, middle), as well as workflows or hints for processing pipelines. FDOs allow us to abstract resources across our hosting sites, thus fostering broader access to data and services. We democratize accessibility and scalability by enabling remote usage and efficient services operation. By integrating FAIR Digital Objects into our data fabric abstraction, we enable standardized, metadata-rich, and cloud-accessible data interactions. This is achieved by embedding relevant metadata, data access instructions, and processing workflows within each object. This allows the system to resolve user queries across storage systems while ensuring the data remains reusable, interoperable, and ready for analysis.

C. Storage and File Formats

The last module of the data fabric abstraction includes the API between the data and the storage. We enhance OpenVisus' functionality as a cloud and caching data model to accelerate data processing. In particular, we tackle the challenge of working with datasets too large for a system's memory by utilizing OpenVisus' out-of-core computations.

Our modules seamlessly exploit file, block, object, and distributed storage options. File storage functions as network-attached storage. Block storage offers high I/O performance, strong consistency, and low-latency connectivity. Object storage guarantees high availability and is durable and infinitely scalable in the cloud. Distributed storage [56]–[58] is tailored for long-term scientific research and is immutable, verifiable, and cost-effective through incentive systems, smart contracts, and quality of service tradeoffs [59]–[61] (Fig. 1, right).

Our abstraction layers allow transparent data conversion between many file formats and optimize storage and retrieval without user intervention. For example, a query for a high-resolution climate model might be stored in Zarr format in the cloud but could be automatically converted to a more compact representation for the user, like GeoTIFF or NetCDF, if required for analysis.

D. Decoupling Data From Storage

Decoupling data from its storage infrastructure is essential for achieving longitudinal data access and sharing capabilities. This separation is crucial because the lifespan of any physical storage medium is inherently shorter than that of the data it holds. Given the rapid evolution of technology and business landscapes, optimal storage solutions can quickly become prohibitively expensive or outdated. Therefore, data repositories must employ technology-agnostic abstractions that facilitate hybrid usage and seamless migration, minimizing costs and disruptions to user access.

Our data fabric abstraction supported visualization framework seamlessly supports: 1) Reading data at varying resolutions within Regions of Interest (ROIs), limiting the result to the available memory or the maximum number of projected pixels on the screen; 2) Generating summary videos from temporal data with specific resolution constraints; the total number of frames is established depending on the network bandwidth and the existence of pre-cached data; 3) Writing multiple versions of data, one for archival at low cost and one resolution-capped for quick sharing purposes.

Furthermore, user requests can be translated to different encoding and compression schemes depending upon several factors, e.g., high-but-slow compression for less frequently accessed data and low-but-fast compression for frequently accessed data. Additionally, our framework facilitates data migration between different storage tiers (hot, warm, and cold) [62] and enables transparent rerouting of data requests from local storage to external storage as needed. To make datasets publicly accessible, we upload petabytes of data after compression to Open Science Data Federation(OSDF) and FTH, an S3-API-compatible decentralized cloud storage and compute service [63].

E. Impact of our Data Fabric Abstraction

Our progressive streaming ability, combined with the cloud-served data in analysis-ready format, allows the user to access and visualize large datasets without downloading the entire file or region. Our frameworks enable convenient remote collaboration and data access with standalone Jupyter notebooks [64] and dashboards using simple lines of code shown in Fig. 3.

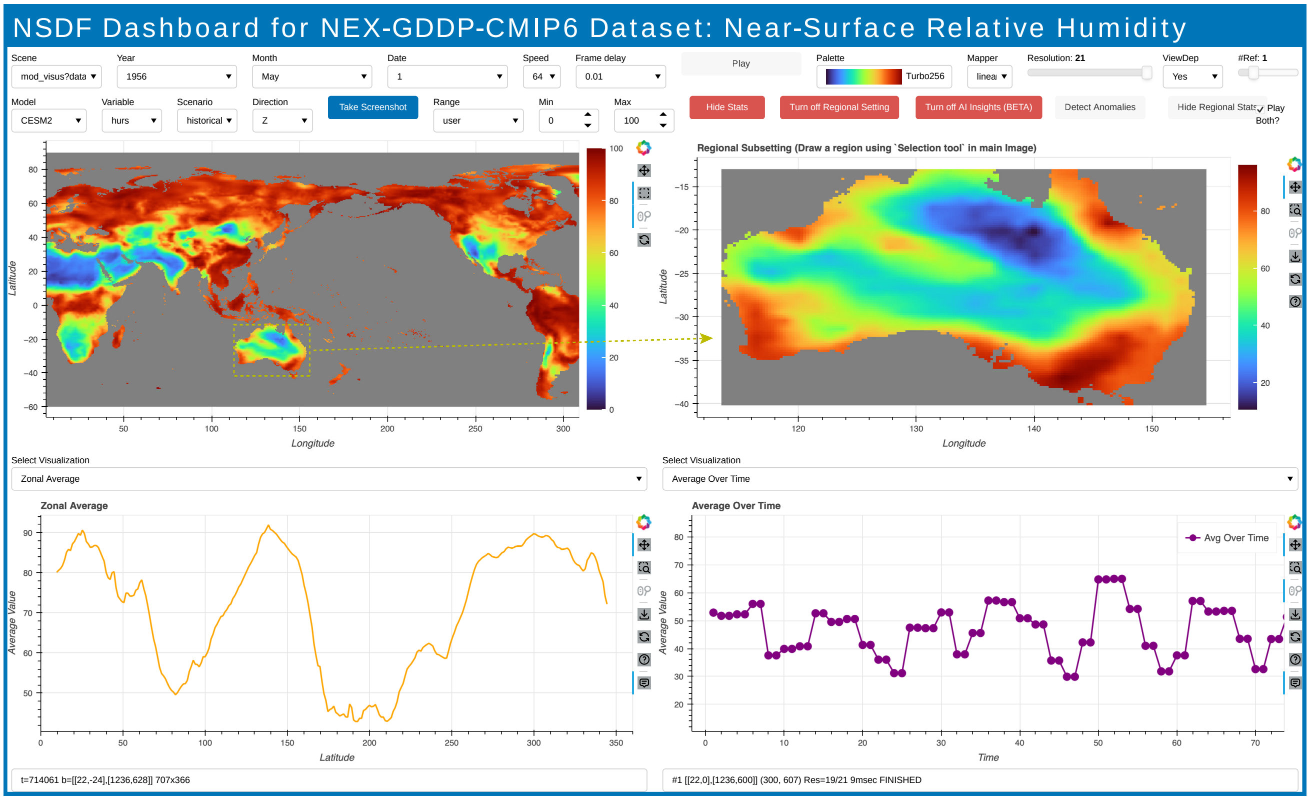
```python
import OpenVisus as ov
dataset="nex-gddp-cmip6"
endpoint="https://atlantis.sci.utah.edu"
url= f"{endpoint}/mod_visus?dataset=/" \
     f"{dataset}&cached=arco"

db=LoadDataset(url)
data=db.read(x=[x_min,x_max],
             y=[y_min,y_max])
```

Fig. 3: Simple Python code fragment for accessing data stored on the Atlantis Server at University of Utah. The result of an input URL given to the ***LoadDataset*** function is assigned to ***db***. The ***db.read*** returns a NumPy array that can easily be used in Python or Jupyter Notebooks.

IV. Dashboards

We have found that, while generating a simulation for a given data set and parameters may require hundreds of hours or more on a supercomputer, viewing those data should no longer require heavy computation. To solve this significant gap between fast reading and massive data visualization, we have integrated a Python version of OpenVisus [33] called OpenVisusPy [65] with web-based visualization frameworks such as Bokeh [66] and Panel [67]. These flexible and interactive widgets support a wide range of visualization techniques, allowing users to dynamically explore, analyze, and understand massive datasets with ease. The data streamed to the browser is a subset of large scientific data, typically in a 2D or 3D NumPy array. Once the data arrive at the client side, users can perform any type of operation or perform analyses themselves. The rendering here is performed on the client side, using interactive Python libraries such as Bokeh and Panel. An integrated environment enables users to interact with their data in previously impractical or resource-intensive ways, allowing data-intensive analysis to become more accessible and insightful. The size of the data transmitted over the network depends on the dataset and the user's requested resolution. For the CMIP6 dataset, users typically view one timestep at a time using a time slider or playback tool. Each timestep for a single model, variable, scenario, and day is originally about 3.5 MB, but after compression, it is reduced to approximately 1 MB for the full-resolution global region. If the user requests lower-resolution data, this can be as small as 512 or even 256 KB



NSDF Dashboard for NEX-GDDP-CMIP6 Dataset: Near-Surface Relative Humidity

Scene: mod_visus?data
Year: 1956
Month: May
Date: 1
Speed: 64
Frame delay: 0.01
Palette: Turbo256
Mapper: linea
Resolution: 21
ViewDep: Yes
#Ref: 1
Model: CESM2
Variable: hurs
Scenario: historical
Direction: Z
Range: user
Min: 0
Max: 100
✓ Play Both?

Regional Subsetting (Draw a region using `Selection tool` in main Image)

Select Visualization: Zonal Average

Select Visualization: Average Over Time

```
t=714061 b=[[22,-24],[1236,628]] 707x366
```

```
#1 [[22,0],[1236,600]] (300, 607) Res=19/21 9msec FINISHED
```

Fig. 4: A Dashboard accessing almost 38 TB of NEX-GDDP-CMIP6 data with Near-Surface Relative Humidity selected. Dashboard contains four views in two columns. The first column provides a global view with user selected Zonal Average in the bottom graph. The right column contains a user selected region with a graph (bottom-right) of average values over time user selected from 1950 to 1956. Other features are histogram, cumulative density function, and inter-modal comparison.

per timestep. Since users can play through time at a rate of up to 10 timesteps per second, the total data streamed can scale to several megabytes per second depending on the interaction. In contrast, data sets such as the LLC2160 are much larger. Each full 3D timestep is approximately 29GB uncompressed and is losslessly compressed to around 8GB. This includes 90 vertical levels, so each level is about 90MB at full resolution, but can be much lower in size for lower resolutions. When a user is viewing a 2D horizontal slice, only a portion of the data is transmitted, which could be a few megabytes per interaction, depending on screen resolution and zoom level. OpenVisus encodes the data using hierarchical z-order space-filling curves, enabling precise and efficient data access [35], [37]. Thus, when a user interacts with the visualization (e.g., slicing through a level), the OpenVisus library [34] sends a targeted request that retrieves only the necessary slice from the server or data source. This approach significantly reduces the amount of data transmitted over the network and allows fast and responsive interactions.

Our dashboard framework, shown in Fig. 2, provides a diverse array of features designed to accommodate both casual explorers and scientific researchers. The user interface of each dashboard is designed by pulling the modular pieces of the framework together, including dataset selection, region of interest extraction, timestep slider, horizontal and vertical slices, color map/palette, colormap range (user or dynamic), resolution sliders, playback functionality, and time speed control, as shown in Fig. 4. The *Show Stats* button gives users the ability to view the general statistics of the data such as average over time, latitudinal average, histogram, and cumulative density function of the data. Furthermore, the machine learning (ML) capabilities integrated within the dashboard empower users to extract actionable insights from their datasets, as will be discussed in Section VI-B3. The ML insights currently supported include anomaly detection, temporal trends, and visualization of deviations from the expected patterns over the entire region, offering a unique approach to interactive data science. Users can also upload a custom ML model before launching the dashboard to extract more value from the dashboard. By enabling users to apply ML techniques directly to their data within the dashboard, we bridge the gap between visualization and computational analysis, significantly reducing the time and expertise needed for scientific discovery.

Our dashboard represents a significant advancement in the visualization and analysis of large-scale data, not limited by the size of the data, the disk space, and the available memory. It allows multiple interactive windows that show streaming progressively loaded slices of volume data, graphs of pixel values through the volume, or macro views of the dataset, as shown in Fig. 4. By providing interactive tools and features, we aim to make complex data visualizations accessible and insightful to a broad audience, from researchers and scientists to educators and policymakers.

V. Custom Volume Renderer

Our current dashboards provide a mapping from data to a 2D color map. We have found that, for some presentations, imagery of 3D volumetric datasets with increased depth perception are crucial. Given the volumetric nature of our datasets, we employed a custom volume renderer to explore their specific aspects, as shown in Fig. 5. Once a region of interest has been found, our dashboards allow users to download a specified region for local rendering.

Our renderer is a GPU-based, ray-traced volume rendering system optimized to handle large datasets in real-time. It is built around the physically-motivated Woodcock (Delta) tracking principle, which homogenizes volumes using fictitious particles derived from the maximum density [68]–[72]. Using this principle the renderer simulates ray-particle collisions by simplifying the probability calculations. Each ray travels until it collides with a non-fictitious particle; at this point, a sample is taken and color-mapped. The probability of a real collision is given by $P_{real}(x) = \frac{\sigma_a(x)}{\bar{\sigma}}$, where $\sigma_a(x)$ represents the exact density at the point x, and $\bar{\sigma}$ denotes the maximum density. After each collision, the renderer calculates the next collision distance by simulating the ray's *free-flight distance* using $t = \frac{-\ln(1-\xi)}{\bar{\sigma}}$, where ξ is a random number. Subdividing the volume into smaller sub-regions with tighter bounds often results in faster ray tracing, a technique we leverage in our implementation. We take one sample per ray to achieve real-time rates. To mitigate the noise, we let the image converge over time by accumulating samples.

We deployed our renderer on a desktop system equipped with an NVIDIA RTX 4090 GPU to deliver high-quality, renderings while providing interactivity. Future updates to our dashboard will include this feature directly within the interface. Currently, the real-time rendering runs on the client side only and is used for producing more stunning imagery after a region of interest has been found in the dashboards.

VI. Examples

Our framework can work for any petascale gridded dataset and throughout this paper, we use three large climate simulation datasets: 1) the NASA 1.8 PB DYAMOND dataset [73], [74], 2) the LLC4320 Ocean dataset [25], [75] and 3) NASA Global Daily Downscaled Projections CMIP6 dataset [76].

In this section, we provide an overview of the datasets and describe four use cases that demonstrate mulitvariate petascale visualization, oceanic and atmospheric variables in a single dashboard, machine-learning powered interactive insights, and dashboards for data democratization for teaching.

A. Dataset Overview

The Coupled Ocean-Atmosphere Simulation (COAS) performed at *NASA Advanced Supercomputing (NAS)* is part of an international project called "Dynamics of the Atmospheric general circulation Modeled On Non-hydrostatic Domains" or DYAMOND. The purpose of COAS is to better understand the oceanic and atmospheric mechanisms that link air-sea interactions with the Earth's water cycle and extreme atmospheric events.

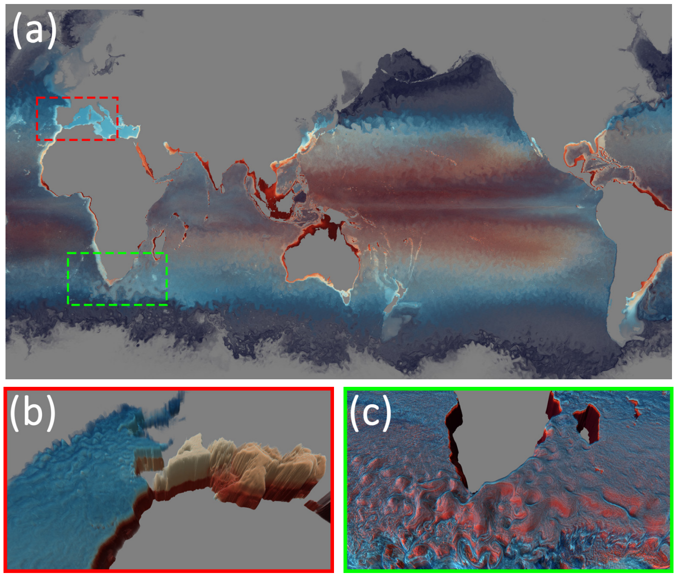


Fig. 5: LLC2160 dataset ocean model volume visualizations (theta/temperature field) in 2560 × 1440 resolution rendered using our Woodcock tracking renderer with different transfer function and camera parameters: (a) an overview of the entire data for a timestep, ≈21 GB, rendered at 180 FPS; (b) a close-up of cool water flowing from the Atlantic Ocean into the warmer Mediterranean Sea, rendered at 190 FPS; and (c) a view of the Cape of Agulhas, highlighting ring-shaped mixing patterns, rendered at 125 FPS.

The first dataset, DYAMOND [73], [74], is the simulation output of research on coupling two models: a global atmospheric model and a global ocean model that were originally designed to be run separately. The atmospheric model is a C1440 configuration of the Goddard Earth Observing System (GEOS) atmospheric model running on a cubed-sphere grid.

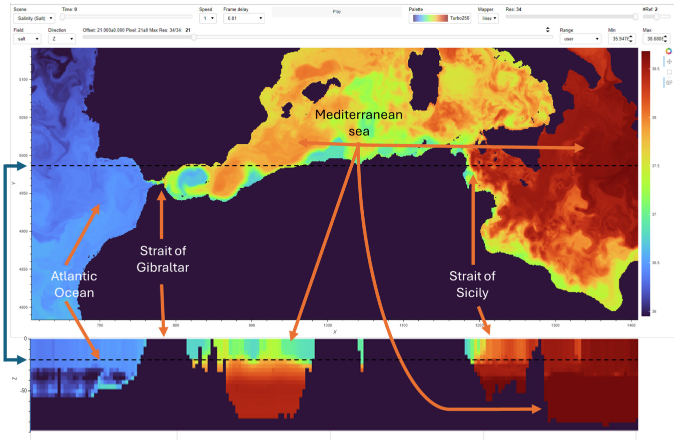

Scene: Salinity (Salt)
Time: 0
Speed: 1
Frame delay: 0.01
Palette: Turbo256
Mapper: linea
Res: 34
#Ref: 2
Field: salt
Direction: Z
Offset: 21.000±0.000 Pixel: 21±0 Max Res: 34/34: 21
Range: user
Min: 35.9476
Max: 38.6806

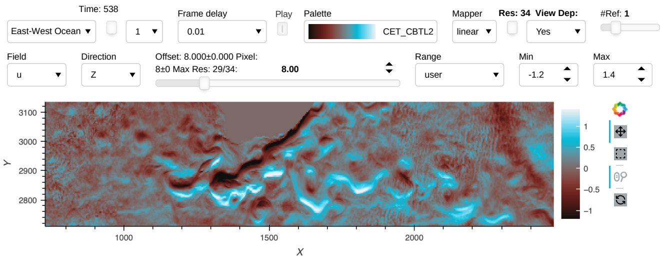
Fig. 6: Case 1: Zoomed-in view of the general water circulation through the Strait of Gibraltar connecting the Mediterranean with the Atlantic Ocean using DYAMOND Data.

The global ocean model is an LLC2160 configuration of the MITgcm (MIT General Circulation Model) model on a lat-lon-cap grid, a Cartesian curvilinear coordinate system approximated with latitude and longitude lines with a cubed-sphere topology. Each model was run for over 10,000 hourly timesteps covering over 14 simulation months. Executing this simulation required almost a full year of computations on nearly 9,000 cores of the Pleiades and Aitken supercomputers at the NAS facility [77]. The atmospheric model output has 20 3D and more than 100 2D scalar fields, and the ocean model output has 5 3D and 15 2D fields. Both models have 3D fields such as temperature, north-south velocity, and east-west velocity. The atmospheric model includes fields such as humidity, soil wetness, snow cover, and various variables of the clouds. The ocean model includes fields of salinity, sea ice thickness, and freshwater flux, with a total size of approximately 1.8 petabytes.

Another ocean dataset, LLC4320 [25], [75], from the 'Estimating the Circulation and Climate of the Ocean' (ECCO) project, is the product of a 14-month simulation of ocean circulation and dynamics using MITgcm model. This simulation is similar to the ocean portion of the DYAMOND coupled simulation but was run with half the horizontal grid spacing (4× the cell count) and with ocean surface boundary values derived from observations and physical models. The model output has five 3D and thirteen 2D fields, including temperature, salinity, three velocity components, sea ice, and radiation. This massive dataset is 2.8 PB.

A third dataset, NEX-GDDP CMIP6, contains 38 TB of daily climate simulation outputs spanning 150 years. This dataset includes variables such as precipitation, air temperature, humidity, and radiation, providing insight into global climate trends and the impact of anthropogenic factors. Unlike DYAMOND and LLC4320, which focus on high-resolution, short-term simulations, NEX-GDDP CMIP6 offers a broader temporal perspective, making it essential for studying long-term climatic changes and variability.

As illustrated in Figs. 6 and 7, as well as shown in the supplementary videos, our dashboards help solve the challenge of putting together all the data, providing access to efficient visualizations in 3D space of multiple atmospheric and oceanic variables. Our dashboards can help improve our understanding of global ocean circulation and its role in Earth's climate system. Built with knowledge from a preexisting animation (Fig. 8), we built dashboards that provide the ability to integrate the DYAMOND and LLC4320 datasets, as shown in Figs. 9 and 10. Our converted datasets of these large-scale simulation datasets are stored in the cloud and on the NAS Pleiades Supercomputer [78]. The instructions for accessing existing deployed dashboards or launch a new one can be accessed from our GitHub repository [79]. We have worked on several GEOS and MITgcm simulation fields, converting them to IDX format, enabling seamless visualization and interaction via Jupyter notebooks and dashboards without intensive computational resources. We also collaborate with several NASA JPL (Jet Propulsion Lab) and NASA ARC (Ames Research Center) scientists to help facilitate the extraction of their region of interest, especially the Gulf Stream and Kuroshio regions (Fig. 9), and have built unique dashboards that display coupled outputs from both GEOS5 and MITgcm configurations of the simulation. Our dashboards combine multiple petascale datasets into a single interface, allowing unprecedented visualization, interaction, and analysis of the massive data. Our integration facilitates a deeper understanding of complex climatic phenomena by enabling scientists to seamlessly navigate and explore data across various scales and dimensions.



Time: 538
East-West Ocean
1
Frame delay: 0.01
Play
Palette: CET_CBTL2
Mapper: linear
Res: 34 View Dep: Yes
#Ref: 1
Field: u
Direction: Z
Offset: 8.000±0.000 Pixel: 8±0 Max Res: 29/34: 8.00
Range: user
Min: -1.2
Max: 1.4

Fig. 7: Formation of Agulhas rings at the African southeast coast demonstrated using the LLC2160 ocean dataset.

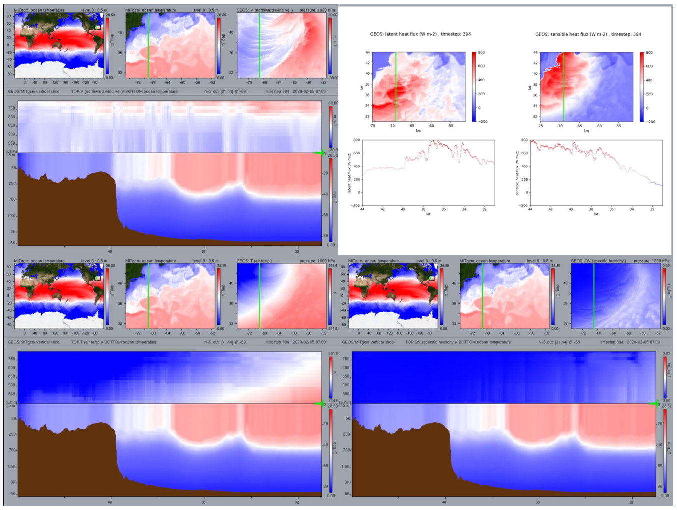


Fig. 8: Prior preliminary animation by Nina McCurdy at NASA ARC created before our collaboration motivated the creation of the dashboard shown in Fig. 9. Image and video courtesy of Nina McCurdy, Copyright NASA 2023.

B. Application-Specific Dashboards

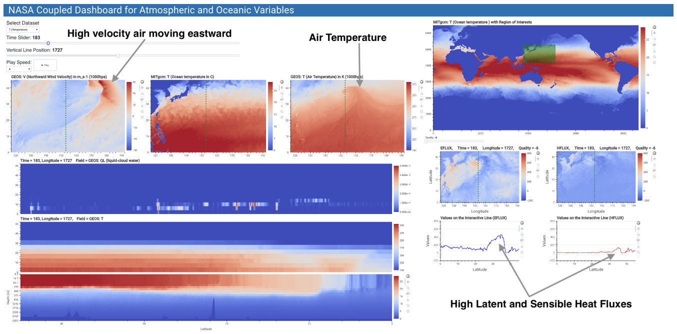
Collaborating with researchers at NASA ARC and NASA JPL, we built several application-specific dashboards to demonstrate our framework. The first two use cases describe a domain expert interaction with feedback. The third use case provides users with ML-powered insights. We also provide a use case with our collaborator at Utah State at Blanding, Native American Serving Non-Tribal Institution (NASNTI).

1) Use Case 1: Multivariate Petascale visualization : To produce useful interactive analysis on massive datasets such as the NASA DYAMOND or LLC4320 Ocean Dataset, visualization scientists typically need to use computing resources at the NAS facility. Although NAS supports and promotes full and open data access to the public, analyzing the data on the supercomputers requires logging into secure platforms and requesting nodes/cores. This dramatically limits the people who may be able to use the data in practice. Scientists and researchers might need to wait hours to days to load the data and produce a video clip for climate scientists, who then perform their scientific tasks on these fixed-resolution animations. Because supercomputing centers often store only full-resolution simulation data, accessing full-domain, full-resolution simulation data can take time, making quick turnaround on-the-fly analysis complex or slow. However, our innovative dashboard and analytical approach significantly streamline working with massive datasets, such as those often encountered in climate science. We reduce the need for extensive computing resources, allowing analysis to be performed on more accessible platforms without compromising accuracy. Users can bypass the inconvenient process of logging into secure platforms and waiting for supercomputer access. Instead, they can directly interact with the data through our user-friendly interface that offers real-time analysis and visualization capabilities.

One key objective of the dashboard shown in Figs. 6 and 7 is to enable the visualization of multiple oceanic variables over time. Traditional visualization techniques struggle with the scale and complexity of the datasets involved, particularly when dealing with simulation outputs spanning 10,000 timesteps across multiple fields. Our dashboard addresses this challenge by offering progressive visualization capabilities and allows scientists to explore the data seamlessly through an intuitive interface [80]. For example, Fig. 6 shows the interesting phenomenon of water circulation around the Mediterranean region. The less saline water from the Atlantic Ocean passes through the Strait of Gibraltar and begins to move eastward. As the water moves east and the evaporation continues, the salinity tends to increase, and the more salty water starts to sink. Oceanographers and climate scientists worldwide have studied this interesting phenomenon, but no tool has ever allowed its interactive illustration on real data to be presented to the general public until now. Another example is around the Agulhas region, as shown in Fig. 7, where warm water from the Agulhas current flows along the southeast coast of South Africa and encounters the colder Atlantic Ocean, which leads the current to bend back on itself. This process, also known as “retroflection,” leads to the formation of large swirling masses of water, creating the Agulhas rings [81]. The dashboard enables any user to examine and interactively explore any regions of interest, as well as play the data across time without waiting for animations to render.

2) Use case 2: Oceanic and atmospheric variables : Our second use case builds on an existing collaboration between visualization researchers at NASA ARC and ocean scientists at JPL/Caltech. The collaboration sought to further investigate the impact of mesoscale and submesoscale (< 500 km) sea-surface temperature anomalies on local atmospheric circulation and vice versa.

Ocean-atmosphere interactions have long been considered to be limited to only the *atmospheric planetary boundary layer (APBL)*, up to 2,000 m above the surface. The ocean surface is modeled as a mixed layer (50 to 200 m deep), with the force of atmosphere acting on the ocean. For example, strong winds deepen the ocean mixed layer, leading to a decrease in the *sea surface temperature (SST)* [82].


NASA Coupled Dashboard for Atmospheric and Oceanic Variables

Select Dataset: T (Temperature)

Time Slider: 183

Vertical Line Position: 1727

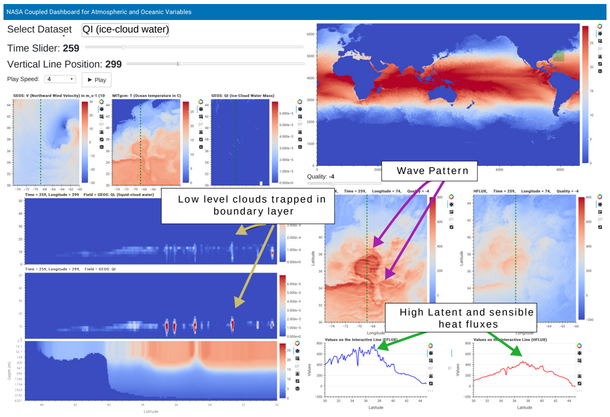
Play Speed: 8

High velocity air moving eastward
Air Temperature
Play
GEOS: V (Northward Wind Velocity) in m_s-1 (1000hpa)
MITgcm: T (Ocean temperature in C)
GEOS: T (Air Temperature) in K (1000hpa)
MITgcm: T (Ocean temperature) with Region of Interests
Quality: -6
Time = 183, Longitude = 1727 Field = GEOS: QL (liquid-cloud water)
Time = 183, Longitude = 1727, Field = GEOS: T
Depth (m)
Latitude
EFLUX, Time = 183, Longitude = 1727, Quality = -6
HFLUX, Time = 183, Longitude = 1727, Quality = -6
Longitude
Values on the Interactive Line (EFLUX)
Values on the Interactive Line (HFLUX)
Values
High Latent and Sensible Heat Fluxes

Fig. 9: Increasing heat fluxes (two plots and images at the bottom right) and air temperature (image at top middle) create a high-velocity wind (image at top left) in the atmosphere moving eastward for the Kuroshio region.

Numerical ocean-atmosphere models coupled with increasing spatial resolutions are challenging previously held theories. It is now known that the ocean-forced variability in the atmosphere is at scales smaller than 500 km, similar to the scales of ocean mesoscale eddies, which constitute up to 80% of the total ocean kinetic energy [82]. Turbulent heat and humidity fluxes are strongly enhanced above warm mesoscale eddies where convection develops and reduces over cold eddies, leading to a significant net heating and humidification of the atmosphere. In addition, this impact of ocean eddies is not confined to the APBL but concerns the whole troposphere, which is up to 12,000 m above sea level. Through these mechanisms, the heat and humidity fluxes associated with ocean eddies intensify atmospheric storms traveling eastward [83] as shown in Fig. 9. As a result, ocean eddies in the Kuroshio-Extension region off Japan can increase precipitations over the West Coast of the U.S.A. by 20% [84]. These results highlight the impact of ocean eddies on the Earth’s water cycle and extreme atmospheric events. Recent studies [85] point to the important role of sea surface temperature (SST) fronts (10 km wide) surrounding ocean eddies on ocean-atmosphere exchanges: SST fronts trigger a secondary circulation, with the same width, in the atmosphere above the APBL that carries heat and humidity to the upper levels.

Over a 6-week period of intermittent collaboration and iteration in Spring 2023, the visualization researchers at NASA ARC and the ocean scientists at JPL/Caltech developed a preliminary visualization showing coupled vertical and horizontal slices of various fields of interest (ocean temperature, air temperature, northward wind velocity, specific humidity, latent heat flux, and sensible heat flux). The visualization, shown in Fig. 8, is an animated version of figures from recently published results [85]. The visualization was highly effective in supporting the ocean scientists’ investigation, leading to important research insights [86], [87], but was limited to vertical and horizontal slices at predefined locations and required the design, development, rendering, and distribution by the visualization researcher. Extracting vertical slices of high-resolution MITgcm data is computationally and I/O intensive due to the native layout of the simulation output. Restrictions


NASA Coupled Dashboard for Atmospheric and Oceanic Variables

Select Dataset QI (ice-cloud water)

Time Slider: 259

Vertical Line Position: 299

Play Speed: 4 ▶ Play

Wave Pattern
Low level clouds trapped in boundary layer
High Latent and sensible heat fluxes
Quality: -4
Latitude

Fig. 10: Interesting wave pattern observed in the plots of sensible and latent heat flux.

on visualization and analysis prompted NASA visualization researchers to develop an high-performance computing (HPC) interactive vertical slicer, leveraging HPC resources (compute nodes, network, storage) at the NAS. Although an effective approach, the HPC vertical slicer requires dedicated compute nodes and dedicated time from visualization researchers. An interactive vertical slicer, accessible to and driven directly by ocean scientists, has been desired without having to be onsite at a supercomputer.

Motivated both by the promising initial results and the limitations of the preliminary animation, the NASA/JPL/Caltech team began collaborating with a team from the Scientific Computing and Imaging Institute at the University of Utah to develop an interactive dashboard version of the preliminary visualization. The collaboration resulted in the dashboard shown in Fig. 10 with the aim of helping to study the relationships between different variables of ocean and atmospheric simulations in different regions of interest, such as the Gulf Stream region and the Kuroshio region. For the Gulf Stream region, we use 75° W to 60° W and 30° N to 45° N. For the Kuroshio region, we use 117° E to 192° E and 0° N to 45° N. The dashboard leverages high-resolution datasets from the GEOS and MITgcm simulations to isolate and visualize the interplay between various climate variables. This region-specific approach allows scientists to observe how atmospheric conditions such as temperature and pressure gradients influence oceanic currents, salinity levels, and vertical velocities and vice versa. Fig. 10 shows how an investigation of an interesting wave pattern observed in the graphs of sensible and latent heat flux (middle) and examination of the associated vertical slices led the ocean scientists to find that the wave pattern was trapped within the atmospheric boundary layer (left) and did not extend above the boundary layer, as previously thought.

3) User Case 3: ML-Powered Interactive Insights: Our NEX-GDDP CMIP6 dashboard represents a significant step forward in enabling climate scientists and researchers to interactively gain actionable insights with the integration of machine learning-based anomaly detection. As shown in Fig. 11, ML-powered interactive insights identify regions with the highest anomalies and temporal trends. This feature can be customized by the users by uploading their own models to the dashboard before launching it. For example, users can instantly identify areas of unusual temperature, precipitation, or humidity, allowing them to focus on potential climate extremes, model biases, or new phenomena. These insights are displayed interactively on the dashboard, with clear visualizations of anomaly intensity and bounding boxes over affected regions.

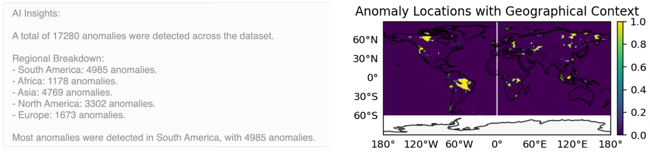
The ML-powered feature, triggered by the "Turn on AI Insights" button, employs a ConvLSTM2D-based model [88] to identify anomalies in climate data over time and space. This model, trained on decades of climate simulations, combines the predictive power of Convolutional Neural Networks (CNNs) for spatial data and Long Short-Term Memory (LSTM) networks for temporal dynamics, making it well suited for analyzing large-scale geospatial datasets.

The model works by leveraging two key inputs: historical climate data sequences and encoded representations of the day of the year (DOY). The input data are normalized to account for seasonal trends and variability. By training the ConvLSTM2D network [88] with sequences of normalized data and corresponding DOY features, the model learns to reconstruct expected climate patterns. Deviations between reconstructed and observed data are marked as anomalies, highlighting regions and times where climate variables differ significantly from historical expectations.

By embedding ML-powered insights into the dashboard, this feature transforms how climate data is analyzed and interpreted. Unlike traditional approaches that require manual examination or offline analysis, this tool automates anomaly detection in real-time, uncovering hidden patterns and outliers directly within scientists' workflows.

In summary, the ML-powered interactive insights feature enhances the NEX-GDDP CMIP6 dashboard's utility, making it not just a visualization platform but a decision-support system. This innovation exemplifies how cutting-edge machine learning techniques can be seamlessly integrated into climate science tools, opening new frontiers for exploration and understanding of Earth's complex systems.

4) Use Case 4: Data Democratization for Teaching : At Utah State Blanding, a Native American Serving Non-Tribal Institution (NASNTI), GEOG 4780/6780 Spatial Analysis is taught by Professor Gustavo Ovando-Montejo. This course, designed for upper division undergraduates and graduates, has 25 students dedicating 4 to 6 hours weekly to spatial analysis using R. The curriculum emphasizes spatial reasoning, coding techniques, and GIScience tasks, including data manipulation, visualization, interpretation, and modeling, with a focus on spatial statistics such as spatial regression.


AI Insights:

A total of 17280 anomalies were detected across the dataset.

Regional Breakdown:
- South America: 4985 anomalies.
- Africa: 1178 anomalies.
- Asia: 4769 anomalies.
- North America: 3302 anomalies.
- Europe: 1673 anomalies.

Most anomalies were detected in South America, with 4985 anomalies.

Fig. 11: Anomalies detected for a random timestep in CMIP6 data. Text shown on the left, regions plotted on the right.

Students in the course face significant challenges with data acquisition, particularly during their final projects, which account for 50% of their grades and involve selecting and analyzing their own data. To assist with this, we provided instructions for installing Jupyter Notebooks and accessing the LLC2160 dataset from the cloud, which includes large fields such as ocean velocities, temperature, and salinity. These datasets are massive, exceeding 1PB in total. Our Jupyter Notebook guides students through loading and visualizing these datasets, and allows them to select specific timesteps and regions for analysis. The first step in the Jupyter Notebook provides an example of loading the Salinity data field with dimensions of 8640*6480*90, with 10366 timesteps. The next step shows users how to load the data and select any timestep and region (x,y,z), at any quality or resolution they want. By Step 4 and within minutes of sitting down with the Jupyter Notebook, the students have a visualization of the NASA data in a plot. Additional steps in the notebook walk the students through querying the data, including calculating the percentage of voxels within the selected salinity range vs. calculating the percentage of world surface within the selected salinity range. The dashboard demonstration showed how students could seamlessly access spatial data stored in the cloud. The students were initially struck by the realization and excitement that they could access these global data and variables in full resolution, akin to methods used by NASA scientists.

The students were excited to focus on specific areas of interest and visualize them in real-time. The dashboard allowed for actual data analysis without downloading, though they could, if desired. A highlight was a spatial query selecting voxels within a range, which students identified as basic suitability analysis — a notable achievement given the data's size. Overall, the dashboards were invaluable for hands-on GIScience teaching.

VII. Discussion

We report data conversion time and compression performance in terms of peak signal-to-noise ratio for the DYAMOND and LLC4320 Ocean datasets. We compare performance across three environments: a native NASA file system, a personal laptop, and a server with cloud data cached to local disks, enabling faster access than reading directly from remote object storage. We also highlight key lessons learned and discuss the broader impact and societal benefits of our dashboards for democratizing access to petascale climate data.

A. Performance for Climate Dashboards

We tested the dashboard objectively in terms of time to read the data at different locations and provide compression metrics using CPU and GPU on Intel hardware.

Time Comparison for Data Processing. To test the efficiency of data conversion and compression techniques, we copied 48 timesteps for a 3D field of the same data, around 1 TB, to different locations, including a NASA supercomputer, a personal computer, and a server. The NAS Pleiades supercomputer with one node and 24 cores took around 1 hour and 20 minutes to convert and an additional 45 minutes to compress the data losslessly. The waiting queue for the job to run was around 10 minutes. The personal computer used for testing was a standard M1 MacBook with 16GB RAM and 8 cores. It took 1 hour 35 minutes to convert the same data and an additional 1 hour 20 minutes to compress it. We also tested the conversion and compression on a 12-core x86-64 architecture Intel Xeon CPU with 64 GB RAM server. Converting the data took 50 minutes on the server and compression took an additional 1 hour 20 minutes. Fig. 12 shows these times in the same graph along with the total time for easier comparison.

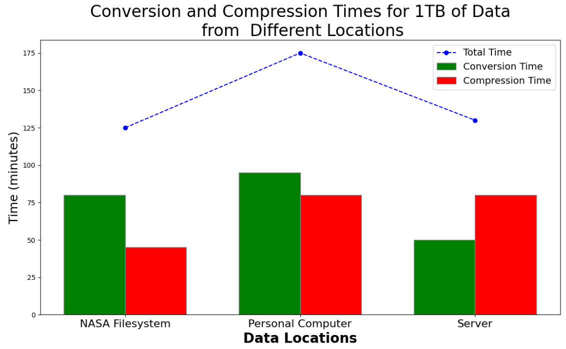
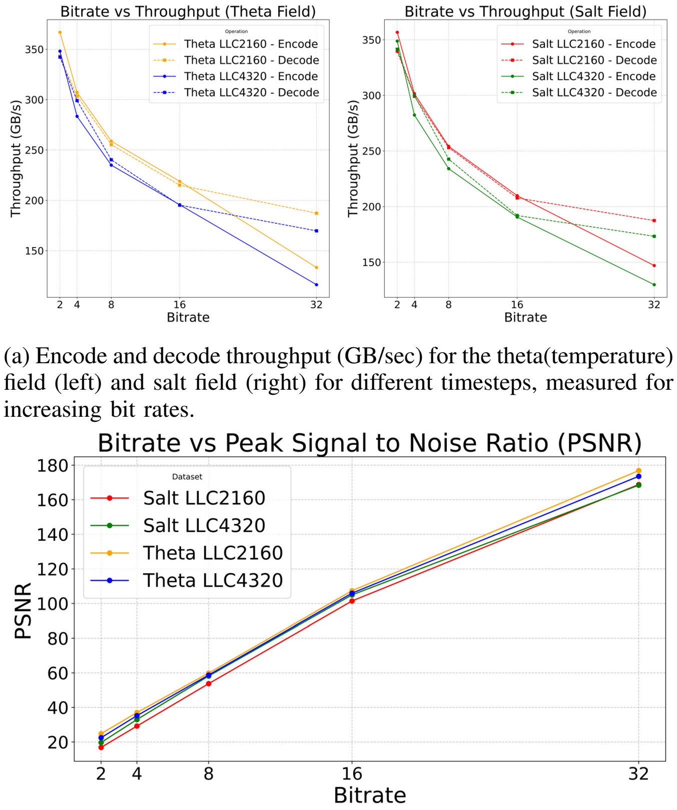


Fig. 12: Experimental results of the time required to convert and compress 1 TB of data from different data locations, such as NASA File System, personal computer, and a server.

Data Compression Performance. To address the challenge of storing and analyzing large high-resolution datasets such as the LLC2160 and LLC4320 datasets, we experimented with compressing some timesteps from both datasets with lossy ZFP compression at different bit rates. We used *peak signal-to-noise ratio (PSNR)* to evaluate the data compression quality of this lossy compression algorithm. PSNR measures the quality of data reconstruction after lossy compression (higher is better). The results show that higher bit precisions, such as 32 and 16 bits, maintain consistently high PSNR across our benchmarks, indicating minimal loss in data quality. In contrast, lower bit rates, such as 2, 4 and 8 bits, result in a significant drop in PSNR, highlighting the trade-off between compression level and data fidelity.

To optimize performance while maintaining cross-platform accessibility, we implemented the ZFP library in SYCL using Intel's oneAPI [89], [90]. We improved the cache and register usage of the ZFP library to fully leverage the GPU's processing power. We evaluated the performance of our GPU implementation of ZFP on Intel Max1550 (Ponte Vecchio) GPUs using multiple timesteps and fields from petascale climate datasets. Figure 13(a) illustrates the throughput (in GB/s) for 3D compression (encoding) and decompression (decoding) at bit rates corresponding to powers of 2, up to 32, for each dataset. Additionally, Figure 13(b) reports the PSNR values of the datasets after decompression. For comparison, we performed the same benchmarks on an Intel Xeon Platinum 8468 CPU using the serial implementation of the ZFP library. Our SYCL-enabled GPU implementation achieved speed-ups ranging from 907× to 564× for compression throughput and from 542× to 329× for decompression throughput compared to the serial CPU implementation. Notably, the highest speedups were observed at higher bit rates, while the lowest speedups occurred at lower bit rates for both compression and decompression.



(a) Encode and decode throughput (GB/sec) for the theta(temperature) field (left) and salt field (right) for different timesteps, measured for increasing bit rates.

(b) Peak signal-to-noise ratio using lossy ZFP compression algorithm for increasing bit rates.

Fig. 13: Throughput and quality evaluation of the ZFP compression algorithm for different datasets. The top row shows the encode and decode throughput (GB/sec) for the theta and salt field across multiple timesteps, measured at increasing bit rates. The bottom row presents the PSNR for the ZFP algorithm at various bit rates.

B. *Lessons Learned*

After several months of intermittent development and iteration, the NASA/JPL/Utah team met (half of the group met in person, half of the group joined remotely) to demonstrate and explore the results. During the session, ocean scientists immediately demonstrated the ability to engage with the data in a way that was not possible before. Discussing phenomena of interest while interacting with the dashboard, they were able to develop new questions/hypotheses and provide preliminary answers about the interaction between the ocean and the atmosphere with a dramatic reduction of cognitive load that only interactive visualization can provide and has always been considered impossible for petascale data without supercomputing resources. Investigation of an interesting wave pattern observed in the plots of sensible and latent heat flux and examination of the associated vertical slices led the ocean scientists to find that the wave pattern was trapped within the atmospheric boundary layer and did not extend above the boundary layer, as previously thought. “This was something quite new for us because we thought it was much above the boundary layer, but no.” The vertical slices in the dashboard shown on the left in Fig. 10 show how the clouds are within the atmospheric boundary layer with very high and wavy latent and sensible heat fluxes.

Our collaboration with oceanic scientists showed the usefulness of faster visualizations and emphasized the need to be able to adjust timesteps and want to be able to view a time-sequence animation with the press of a button, which was crucial to seeing the expected correlation between multiple fields. Because each visualization generated in this interactive environment could take days to generate using the status quo fixed-frame animation approach, the domain scientists commented that our on-the-fly and interactive visualization “was astonishing.” The ocean scientists also noted that the ability to interactively adjust the vertical slice location allowed them to check for numerical instabilities near the vertical cuts, a capability they previously lacked. Additionally, the interactive dashboard reduced the barrier to exploration and collaboration by enabling investigations based on real-time data observations, facilitating more dynamic and immediate scientific inquiry.

Future iterations will help the new visualization dashboards better match the traditional workflow of domain scientists. We need to be able to automate the output of animations for high-resolution data. However, as our use cases show, being able to set the resolution low for low bandwidth or quick investigations is something that climate scientists and visualization researchers could not easily do. The ability to share with college students the same data NASA and JPL scientists use for their research is creating an invaluable educational bridge. By enabling this accessibility, we are not only democratizing access to the latest scientific data but also inspiring the next generation of scientists and data analysts to explore the climate phenomena with the same tools used by leading researchers.

VIII. Conclusion

Our unified framework not only addresses the challenges of petascale data visualization and analysis but also embodies the FAIR principles, enhancing scientific data democratization. Our framework is part of a larger movement to make petascale climate data easily *Findable* [91] [92], enabling users to easily discover and locate relevant datasets. Data are made *Accessible* via publicly available Web links to cloud-stored optimized formats, allowing researchers worldwide to access petascale data without the need for specialized infrastructure. The framework ensures *Interoperability* through its novel data fabric abstraction layer that includes the use of standard protocols and APIs, integrating multiple software components and analysis tools. This allows for seamless integration with existing scientific workflows and tools. The use of Analysis-Ready Cloud-Optimized (ARCO) formats further enhances interoperability by providing standardized, cloud-friendly data structures. *Reusability* is promoted through comprehensive metadata management, which is a cornerstone of our data fabric approach. This metadata not only facilitates data discovery, but also provides context for proper data interpretation and reuse. Additionally, our framework's workflow ensures that analyses are easily reproducible and reusable by other researchers [92]. By adhering to these FAIR principles, our unified framework enables researchers from diverse scientific domains and institutions to access, analyze, and visualize petascale data. The democratization of data and analytical tools has the potential to accelerate scientific discovery and foster collaborative research on a global scale.

IX. Github link for code and supplementals

https://github.com/sci-visus/Openvisus-NASA-Dashboard

Acknowledgements

The authors wish to thank the NASA Ames Research Center, NASA Jet Propulsion Lab, Future Tech Holdings (https://future-tech-holdings.com/), Open Science Data Federation (OSDF, https://osg-htc.org/services/osdf.html) and ViSOAR LLC (https://visoar.com). NASA ARC scientists, Nina McCurdy and David Ellsworth, provided relevant usecases, and NASA HECC provided data and computational resources via the Pleiades supercomputer. FTH and OSDF-Pelican provided cloud storage to host the data. ScientistCloud by ViSOAR LLC provided interface to dashboards via their secure server. This work was funded in part by NASA Cooperative Agreement 80NSSC23M0013, NASA JPL Subcontract No. 1685389, DOE SBIR Phase II #DE-SC0017152, NSF OAC award 2138811, NSF CI CoE Award 2127548, NSF OISE award 2330582, DoE award DE-FE0031880, the Intel oneAPI Center of Excellence at University of Utah. Results presented in this paper were obtained in part using the Chameleon, Cloudlab, CloudBank, Fabric, and ACCESS testbeds supported by the National Science Foundation.

References

[1] NASA, "GEOS/ECCO Coupled Nature Run Data Portal," https://data.nas.nasa.gov/geosecco/geoseccodata/c1440_llc2160/, 2021, accessed: Mar 29, 2024.

[2] ——, "ECCO Data Portal," 2014, accessed: Mar 29, 2024.

[3] M. Cox and D. Ellsworth, "Managing big data for scientific visualization," 01 1997.

[4] J. D. Ford, S. E. Tilleard, L. Berrang-Ford, M. Araos, R. Biesbroek, A. C. Lesnikowski, G. K. MacDonald, A. Hsu, C. Chen, and L. Bizikova, "Big data has big potential for applications to climate change adaptation," *Proceedings of the National Academy of Sciences*, vol. 113, no. 39, pp. 10 729–10 732, 2016.

[5] K. Team, "Keras documentation: Convlstm2d layer." [Online]. Available: https://keras.io/api/layers/recurrent_layers/conv_lstm2d/

[6] M. Bostock, V. Ogievetsky, and J. Heer, "D^3 Data-Driven Documents," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2301–2309, 2011.

[7] Y. Wang, "Deck.gl: Large-scale web-based visual analytics made easy," *arXiv preprint arXiv:1910.08865*, 2019.

[8] E. Angel and E. Haines, "An interactive introduction to WEBGL and three. JS," in *ACM SIGGRAPH 2017 Courses*, 2017, pp. 1–95.

[9] W. Usher and V. Pascucci, "Interactive visualization of terascale data in the browser: Fact or fiction?" in *2020 IEEE 10th Symposium on Large Data Analysis and Visualization (LDAV)*. IEEE, 2020, pp. 27–36.

[10] W. Usher, L. Dyken, and S. Kumar, "Speculative Progressive Raycasting for Memory Constrained Isosurface Visualization of Massive Volumes," in *2023 IEEE 13th Symposium on Large Data Analysis and Visualization (LDAV)*, 2023, pp. 1–11.

[11] J. R. Alder and S. W. Hostetler, "Web based visualization of large climate data sets," *Environmental Modelling & Software*, vol. 68, pp. 175–180, 2015.

[12] J. D. Walker, B. H. Letcher, K. D. Rodgers, C. C. Muhlfeld, and V. S. D'Angelo, "An Interactive Data Visualization Framework for Exploring Geospatial Environmental Datasets and Model Predictions," *Water*, vol. 12, no. 10, 2020. [Online]. Available: https://www.mdpi.com/2073-4441/12/10/2928

[13] S. Jourdain, U. Ayachit, and B. Geveci, "ParaViewWeb: A web framework for 3D visualization and data processing," *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 3, no. 1, pp. 870–877, 2011.

[14] M. Raji, A. Hota, T. Hobson, and J. Huang, "Scientific visualization as a microservice," *IEEE transactions on visualization and computer graphics*, vol. 26, no. 4, pp. 1760–1774, 2018.

[15] S. Lu, R. M. Li, W. C. Tjhi, K. K. Lee, L. Wang, X. Li, and D. Ma, "A Framework for Cloud-Based Large-Scale Data Analytics and Visualization: Case Study on Multiscale Climate Data," in *2011 IEEE Third International Conference on Cloud Computing Technology and Science*, 2011, pp. 618–622.

[16] R. Rajatheva, "Performance Challenges with Data Visualizations in Browser Environment," 2023.

[17] A. B. Gurvich and A. M. Geller, "Firefly: A Browser-based Interactive 3D Data Visualization Tool for Millions of Data Points," *The Astrophysical Journal Supplement Series*, vol. 265, no. 2, p. 38, mar 2023. [Online]. Available: https://dx.doi.org/10.3847/1538-4365/acb59f

[18] K. Begum, M. M. Rashid, and M. A. U. Shariff, "Comparative Study of Big Data Visualization Tools and Techniques," in *Applied Informatics for Industry 4.0*. Chapman and Hall/CRC, 2023, pp. 188–199.

[19] D. N. Williams, T. Bremer, C. Doutriaux, J. Patchett, S. Williams, G. Shipman, R. Miller, D. R. Pugmire, B. Smith, C. Steed, E. W. Bethel, H. Childs, H. Krishnan, P. Prabhat, M. Wehner, C. T. Silva, E. Santos, D. Koop, T. Ellqvist, J. Poco, B. Geveci, A. Chaudhary, A. Bauer, A. Pletzer, D. Kindig, G. L. Potter, and T. P. Maxwell, "Ultrascale Visualization of Climate Data," *Computer*, vol. 46, no. 9, 2013.

[20] H. Aizenman, M. Grossberg, D. Jones, N. Barnes, J. Smerdon, K. Anchukaitis, and J. E. Geay, "Web Based Visualization Tool for Climate Data Using Python," in *92nd AMS Annual Meeting, Second Symposium on Advances in Modeling and Analysis Using Python. Sl: sn*, 2012.

[21] X. Sun, S. Shen, G. G. Leptoukh, P. Wang, L. Di, and M. Lu, "Development of a Web-based visualization platform for climate research using Google Earth," *Computers & Geosciences*, vol. 47, pp. 160–168, 2012.

[22] L. Zepner, P. Karrasch, F. Wiemann, and L. Bernard, "ClimateCharts.net–an interactive climate analysis web platform," *International Journal of Digital Earth*, vol. 14, no. 3, pp. 338–356, 2021.

[23] P. C. Wong, H.-W. Shen, R. Leung, S. Hagos, T.-Y. Lee, X. Tong, and K. Lu, "Visual analytics of large-scale climate model data," in *2014 IEEE 4th Symposium on Large Data Analysis and Visualization (LDAV)*, 2014, pp. 85–92.

[24] P. C. Wong, H.-W. Shen, and C. Chen, "Top ten interaction challenges in extreme-scale visual analytics," *Expanding the frontiers of visual analytics and visualization*, pp. 197–207, 2012.

[25] I. Fukumori, O. Wang, I. Fenty, G. Forget, P. Heimbach, and R. M. Ponte, "ECCO Version 4 Release 3," 2017.

[26] R. Abernathey, "Petabytes of Ocean Data, Part 1: NASA Ecco Data Portal," *Medium*, Jun 2019.

[27] A. Barrett, C. Battisto, B. Bottomley, A. Friesz, A. Hunzinger, M. Jami, A. Lewandowski, B. Lind, L. López, J. McNelis, and et al., "NASA EarthData Cloud Cookbook," *Zenodo*, May 2023.

[28] D. Hoang, A. Panta, G. Scorzelli, P. Davis, M. Parashar, and V. Pascucci, "Publishing NASA's multi-petabytes of climate datasets," Dec 2022. [Online]. Available: https://doi.org/10.5281/zenodo.7488835

[29] xmitgcm, "Xmitgcm," https://xmitgcm.readthedocs.io/en/latest/.

[30] R. P. Abernathey, T. Augspurger, A. Banihirwe, C. C. Blackmon-Luca, T. J. Crone, C. L. Gentemann, J. J. Hamman, N. Henderson, C. Lepore, T. A. McCaie *et al.*, "Cloud-native repositories for big scientific data," *Computing in Science & Engineering*, vol. 23, no. 2, pp. 26–35, 2021.

[31] NASA, "SOTO by Worldview," https://soto.podaac.earthdatacloud.nasa.gov/.

[32] D. A. Ellsworth, C. E. Henze, and B. C. Nelson, "Interactive visualization of high-dimensional petascale ocean data," in *2017 IEEE 7th Symposium on Large Data Analysis and Visualization (LDAV)*, 2017, pp. 36–44.

[33] V. Pascucci, G. Scorzelli, B. Summa, P.-T. Bremer, A. Gyulassy, C. Christensen, S. Philip, and S. Kumar, "The ViSUS visualization framework," in *High Performance Visualization*. Chapman and Hall/CRC, 2012, pp. 439–452.

[34] OpenVisus, "OpenVisus," https://github.com/sci-visus/openvisus.

[35] B. Summa, G. Scorzelli, M. Jiang, P.-T. Bremer, and V. Pascucci, "Interactive editing of massive imagery made simple: Turning Atlanta into Atlantis," *ACM Trans. on Graphics (TOG)*, vol. 30, no. 2, 2011.

[36] S. Kumar, S. Petruzza, W. Usher, and V. Pascucci, "Spatially-Aware Parallel I/O for Particle Data," in *Proceedings of the 48th International Conference on Parallel Processing*, ser. ICPP 2019. New York, NY, USA: Association for Computing Machinery, 2019.

[37] S. Kumar, C. Christensen, J. A. Schmidt, P. Bremer, E. Brugger, V. Vishwanath, P. H. Carns, H. Kolla, R. W. Grout, J. Chen, M. Berzins, G. Scorzelli, and V. Pascucci, "Fast Multiresolution Reads of Massive Simulation Datasets," in *29th International Supercomputing Conference, ISC 2014, Leipzig, Germany, June 22-26, 2014. Proceedings*, ser. Lecture Notes in Computer Science, vol. 8488. Springer, 2014.

[38] W. Usher, X. Huang, S. Petruzza, S. Kumar, S. R. Slattery, S. T. Reeve, F. Wang, C. R. Johnson, and V. Pascucci, "Adaptive Spatially Aware I/O for Multiresolution Particle Data Layouts," in *2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, 2021.

[39] L. Grandinetti, G. R. Joubert, M. Kunze, and V. Pascucci, Eds., *Big Data and High Performance Computing - Selected Papers from HPC Workshop, Cetraro, Italy, July 7-11, 2014*, ser. Advances in Parallel Computing, vol. 26. IOS Press, 2015. [Online]. Available: http://ebooks.iospress.nl/volume/big-data-and-high-performance-computing

[40] A. Venkat, C. Christensen, A. Gyulassy, B. Summa, F. Federer, A. Angelucci, and V. Pascucci, "A Scalable Cyberinfrastructure for Interactive Visualization of Terascale Microscopy Data," *... New York Scientific Data Summit (NYSDS) : proceedings ...*, vol. 2016, 08 2016.

[41] V. Pascucci and R. J. Frank, "Global static indexing for real-time exploration of very large regular grids," in *Proceedings of the 2001 ACM/IEEE conference on Supercomputing, Denver, CO, USA, November 10-16, 2001, CD-ROM*, G. Johnson, Ed. ACM, 2001, p. 2.

[42] S.-E. Yoon, P. Lindstrom, V. Pascucci, and D. Manocha, "Cache-oblivious mesh layouts," *ACM Transactions on graphics: ACM SIGGRAPH 2005 Papers*, vol. 24, no. 3, pp. 886–893, 2005. [Online]. Available: http://doi.acm.org/10.1145/1073278

[43] D. Hoang, B. Summa, H. Bhatia, P. Lindstrom, P. Klacansky, W. Usher, P.-T. Bremer, and V. Pascucci, "Efficient and Flexible Hierarchical Data Layouts for a unified encoding of scalar field precision and resolution," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 603–613, 2021.

[44] H. Bhatia, D. Hoang, N. Morrical, V. Pascucci, P.-T. Bremer, and P. Lindstrom, "AMM: Adaptive Multilinear Meshes," *IEEE Trans. on Visualization and Computer Graphics*, vol. 28, no. 6, 2022.

[45] D. Hoang, P. Klacansky, H. Bhatia, P.-T. Bremer, P. Lindstrom, and V. Pascucci, "A Study of the Trade-off Between Reducing Precision and Reducing Resolution for Data Analysis and Visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 1193–1203, 2019.

[46] A. Panta, A. Gooch, G. Scorzelli, M. Taufer, and V. Pascucci, "Scalable climate data analysis: Balancing petascale fidelity and computational cost," in *2025 IEEE 25th International Symposium on Cluster, Cloud and Internet Computing Workshops (CCGridW)*, 2025, pp. 245–248.

[47] C. Stern, R. Abernathey, J. Hamman, R. Wegener, C. Lepore, S. Harkins, and A. Merose, "Pangeo forge: crowdsourcing analysis-ready, cloud optimized data production," *Frontiers in Climate*, vol. 3, p. 782909, 2022.

[48] S. Hoyer and J. Hamman, "XArray: N-D labeled Arrays and Datasets in Python," *Journal of Open Research Software*, vol. 5, no. 1, p. 10, Apr. 2017. [Online]. Available: http://dx.doi.org/10.5334/jors.148

[49] S. Klasky, J. Thayer, and H. Najm, "Data Reduction for Science: Brochure from the Advanced Scientific Computing Research Workshop," Apr. 2021. [Online]. Available: http://dx.doi.org/10.2172/1770192

[50] N. R. Council, *Frontiers in Massive Data Analysis*. Washington, DC: The National Academies Press, 2013. [Online]. Available: https://nap.nationalacademies.org/catalog/18374/frontiers-in-massive-data-analysis

[51] N. G. Author, "Synergistic Challenges in Data-Intensive Science and Exascale Computing. Summary report of the Advanced Scientific Computing Advisory Committee (ASCAC) Subcommittee, March 2013," Mar. 2013. [Online]. Available: http://dx.doi.org/10.2172/1471113

[52] G. Strawn, "Open Science, Business Analytics, and FAIR Digital Objects," in *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, vol. 2, 2019, pp. 658–663.

[53] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne *et al.*, "The FAIR Guiding Principles for scientific data management and stewardship," *Scientific data*, vol. 3, no. 1, 2016.

[54] M. Taufer, H. Martinez, J. Luettgau, L. Whitnah, G. Scorzelli, P. Newell, A. Panta, P.-T. Bremer, D. Fils, C. R. Kirkpatrick, and V. Pascucci, "Enhancing scientific research with fair digital objects in the national science data fabric," *Computing in Science & Engineering*, vol. 25, no. 5, pp. 39–47, Sep. 2023.

[55] R. O. of Belgium, "Fair-gnss," https://fair-gnss.oma.be/method.php.

[56] A. Ismail, M. Toohey, Y. C. Lee, Z. Dong, and A. Y. Zomaya, "Cost and Performance Analysis on Decentralized File Systems for Blockchain-Based Applications: State-of-the-Art Report," in *2022 IEEE International Conference on Blockchain (Blockchain)*, 2022.

[57] E. Daniel and F. Tschorsch, "IPFS and friends: A qualitative comparison of next generation peer-to-peer data networks," *CoRR*, 2021. [Online]. Available: https://arxiv.org/abs/2102.12737

[58] H. Kotabe, "Decentralized Storage: A Primer," https://www.tbstat.com/wp/uploads/2022/05/20220531_DecentralizedStorage_TheBlockResearch.pdf, May 2022, accessed: 31/03/2024.

[59] X. Li, Q. Liu, S. Wu, Z. Cao, and Q. Bai, "Game theory based compatible incentive mechanism design for non-cryptocurrency blockchain systems," *J. of Industrial Information Integration*, vol. 31, p. 100426, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2452414X22000930

[60] I. Vakilinia, W. Wang, and J. Xin, "An Incentive-Compatible Mechanism for Decentralized Storage Network," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 4, p. 2294–2306, Jul. 2023. [Online]. Available: http://dx.doi.org/10.1109/TNSE.2023.3245326

[61] J. Shen, Y. Li, Y. Zhou, and X. Wang, "Understanding I/O performance of ipfs storage: a client's perspective," in *Proceedings of the International Symposium on Quality of Service*, ser. IWQoS '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: https://doi.org/10.1145/3326285.3329052

[62] C. Pernet, C. Svarer, R. Blair, J. D. V. Horn, and R. A. Poldrack, "On the long-term archiving of research data," 2023.

[63] "Future tech holdings," Aug 2024. [Online]. Available: https://future-tech-holdings.com/

[64] B. M. Randles, I. V. Pasquetto, M. S. Golshan, and C. L. Borgman, "Using the Jupyter notebook as a tool for open science: An empirical study," in *2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*. IEEE, 2017, pp. 1–2.

[65] OpenVisus, "OpenVisuspy," https://github.com/sci-visus/openvisuspy.

[66] Bokeh, https://bokeh.org/, bokeh.

[67] Holoviz, "Panel: The Powerful Data Exploration and Web App Framework for python," https://github.com/holoviz/panel.

[68] E. Woodock, T. Murphy, H. P., and L. T.C., "Techniques used in the GEM Code for Monte Carlo Neutronics Calculation in Reactors and Other Systems of Complex Geometry," Argonne National Laboratory, Tech. Rep., 1965.

[69] M. Raab, D. Seibert, and A. Keller, "Unbiased Global Illumination with Participating Media," in *Monte Carlo and Quasi-Monte Carlo Methods 2006*, 2008.

[70] J. Novák, A. Selle, and W. Jarosz, "Residual ratio tracking for estimating attenuation in participating media," *ACM Trans. Graph.*, vol. 33, no. 6, 2014.

[71] N. Morrical, A. Sahistan, U. Güdükbay, I. Wald, and V. Pascucci, "Quick Clusters: A GPU-Parallel Partitioning for Efficient Path Tracing of Unstructured Volumetric Grids," *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, no. 01, 2023.

[72] A. Sahistan, S. Zellmann, N. Morrical, V. Pascucci, and I. Wald, "Multi-Density Woodcock Tracking: Efficient & High-Quality Rendering for Multi-Channel Volumes," in *Eurographics Symposium on Parallel Graphics and Visualization*. The Eurographics Association, 2025.

[73] B. Stevens, M. Satoh, L. Auger, J. Biercamp, C. S. Bretherton, X. Chen, P. Düben, F. Judt, M. Khairoutdinov, D. Klocke *et al.*, "DYAMOND: the DYnamics of the Atmospheric general circulation Modeled On Non-hydrostatic Domains," *Progress in Earth and Planetary Science*, vol. 6, no. 1, pp. 1–17, 2019.

[74] NASA, "DYnamics of the Atmospheric general circulation Modelled On Non-hydrostatic Domains Phase," https://gmao.gsfc.nasa.gov/global_mesoscale/dyamond_phaseII/data_access/.

[75] D. Menemenlis, C. Hill, C. Henze, J. Wang, and I. Fenty, "Pre-SWOT Level-4 Hourly MITgcm LLC4320 Native 2km Grid Oceanographic Version 1.0," 2021.

[76] B. Thrasher, W. Wang, A. Michaelis, F. Melton, T. Lee, and R. Nemani, "Nasa global daily downscaled projections, cmip6," *Scientific Data*, vol. 9, no. 1, Jun 2022.

[77] NASA, "A global, coupled ocean-atmosphere simulation with kilometer-scale resolution." [Online]. Available: https://www.nas.nasa.gov/SC21/research/project16.html

[78] N. A. S. Division, "NASA High-End Computing Capability," https://www.nas.nasa.gov/hecc/.

[79] Sci-Visus, "Interactive visualization of petscale climate data using OpenVisus," https://github.com/sci-visus/Openvisus-NASA-Dashboard.

[80] A. Panta, X. Huang, N. McCurdy, D. Ellsworth, A. A. Gooch, G. Scorzelli, H. Torres, P. Klein, G. A. Ovando-Montejo, and V. Pascucci, "Web-based visualization and analytics of petascale data: Equity as a tide that lifts all boats," in *2024 IEEE 14th Symposium on Large Data Analysis and Visualization (LDAV)*, 2024, pp. 1–11.

[81] D. B. Olson and R. H. Evans, "Rings of the Agulhas current," *Deep Sea Research Part A. Oceanographic Research Papers*, vol. 33, no. 1, pp. 27–42, 1986.

[82] C. L. Gentemann, C. A. Clayson, S. Brown, T. Lee, R. Parfitt, J. T. Farrar, M. Bourassa, P. J. Minnett, H. Seo, S. T. Gille, and V. Zlotnicki, "FluxSat: Measuring the Ocean–Atmosphere Turbulent Exchange of Heat and Moisture from Space," *Remote Sensing*, vol. 12, no. 11, 2020. [Online]. Available: https://www.mdpi.com/2072-4292/12/11/1796

[83] A. Foussard, G. Lapeyre, and R. Plougonven, "Response of Surface Wind Divergence to Mesoscale SST Anomalies under Different Wind Conditions," *J. of the Atmospheric Sciences*, vol. 76, pp. 2065 – 2082, 2019. [Online]. Available: https://journals.ametsoc.org/view/journals/atsc/76/7/jas-d-18-0204.1.xml

[84] X. Liu, X. Ma, P. Chang, Y. Jia, D. Fu, G. Xu, L. Wu, R. Saravanan, and C. M. Patricola, "Ocean fronts and eddies force atmospheric rivers and heavy precipitation in western North America," *Nature Communications*, vol. 12, no. 1, p. 1268, 2021.

[85] E. Strobach, P. Klein, A. Molod, A. A. Fahad, A. Trayanov, D. Menemenlis, and H. Torres, "Local Air-Sea Interactions at Ocean Mesoscale and Submesoscale in a Western Boundary Current," *Geophysical Research Letters*, 03 2022.

[86] F. Vivant, L. Siegelman, P. Klein, H. Torres, D. Menemenlis, and A. Molod, "Ocean submesoscale fronts induce diabatic heating and convective precipitation within storms," *Communications Earth & Environment*, vol. 6, p. 1234567890, 01 2025.

[87] H. Torres, P. Klein, L. Siegelman, F. Vivant, D. Menemenlis, A. Molod, E. Strobach, and N. McCurdy, "Seasonality of air-sea coupling through submesoscale ocean gradients and latent heat fluxes in a Western Boundary Current," *In preparation: Geophysical Research Letters*, 2024.

[88] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[89] Khronos SYCL Working Group, "Sycl specification," https://www.khronos.org/sycl/, 2025, 2020 Specification.

[90] Intel Corporation, "oneapi data parallel c++," https://www.intel.com/content/www/us/en/developer/tools/oneapi/data-parallel-c-plus-plus.html, 2024, accessed: 2024-07-07.

[91] J. Luettgau, G. Scorzelli, V. Pascucci, G. Tarcea, C. R. Kirkpatrick, and M. Taufer, "Nsdf-catalog: Lightweight indexing service for democratizing data delivery," in *2022 IEEE/ACM 15th International Conference on Utility and Cloud Computing (UCC)*, 2022, pp. 1–10.

[92] M. Taufer, H. Martinez, A. Panta, P. Olaya, J. Marquez, A. Gooch, G. Scorzelli, and V. Pascucci, "Leveraging national science data fabric services to train data scientists," in *SC24-W: Workshops of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 2024, pp. 355–362.

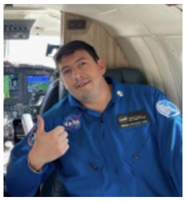
Aashish Panta is a graduate student at the University of Utah, working with the CEDMAV group at the Scientific Computing and Imaging (SCI) Institute. His research interests include large-scale data management, visualization techniques, and leveraging machine learning algorithms on large datasets. He has contributed to several interdisciplinary projects focused on improving the efficiency and scalability of scientific data workflows, and collaborated with domain scientists to develop user-friendly visualization tools.

Alper Sahistan received his B.S. and M.Sc. degrees in Computer Engineering from Bilkent University in 2019 and 2022, respectively. He is pursuing a Ph.D. at the University of Utah, researching high-performance scientific visualization, GPU-based ray tracing, and compressed representations of large scientific data with the CEDMAV Group at the SCI Institute. His work aims to enable efficient and interactive exploration of multimodal scientific data on modern computing architectures.

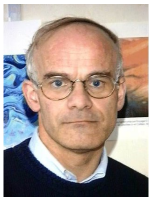
Xuan Huang is a Graphics Software Engineer at Cesium, working on efficient spatial data processing and visualization pipelines for large geospatial datasets. She completed her PhD degree in computing from the Scientific Computing and Imaging Institute at the University of Utah, where she conducted research under the supervision of Valerio Pascucci. Her research interests focused on scalable scientific data visualization systems and distributed computing for large-scale data.

Amy A. Gooch is the Chief Operating Officer at ViSOAR LLC and an affiliate of the University of Utah's Scientific Computing and Imaging Institute. In her leadership role, she manages the delivery of cutting-edge storage-as-a-service solutions that support a wide range of scientific and research applications. Amy plays a vital role in advancing projects focused on large-scale data management and visualization, helping to bridge the gap between innovative research and practical technology deployment. Her expertise contributes significantly to enabling researchers to efficiently handle and analyze complex datasets, fostering collaboration and discovery across multiple scientific disciplines.

Giorgio Scorzelli is the Director of Software Development at the Center for Extreme Data Management Analysis and Visualization (CEDMAV) within the Scientific Computing and Imaging (SCI) Institute at the University of Utah. He is recognized as an expert in software engineering, particularly in transitioning research into production-ready software, and has played a key role as a principal software design engineer and co-PI on NSF/DOE projects focused on large-scale imaging technologies. He is also a core contributor to the OpenVisus platform and serves as Director of Software Development for the National Science Data Fabric (NSDF). Scorzelli holds several industry certifications, including AWS Certified Solutions Architect and Google Cloud Certified Associate Cloud Engineer.



Hector Torres is a researcher at the NASA Jet Propulsion Laboratory (JPL) in Pasadena, California. His work focuses on ocean circulation and the interaction between the ocean and atmosphere. He utilizes a combination of in-situ, satellite, and airborne observations, along with numerical models, to study ocean dynamics across a range of spatial scales. Recent research emphasizes understanding how the ocean influences the atmosphere on sub-seasonal timescales and at scales smaller than hundreds of kilometers. Dr. Torres has contributed to several international collaborations aimed at advancing the predictive understanding of coupled ocean-atmosphere processes. He is also actively involved in mentoring early-career scientists and developing innovative observational strategies for future Earth science missions.



Patrice Klein is a Visiting Associate in Environmental Science and Engineering at the Division of Geological and Planetary Sciences, California Institute of Technology (Caltech). His research focuses on environmental science and engineering, and he is affiliated with both Caltech and the NASA Jet Propulsion Laboratory in Pasadena, California. Dr. Klein is internationally recognized for his pioneering work on oceanic turbulence, mesoscale and submesoscale dynamics, and their impact on global climate and marine ecosystems. He has authored numerous influential publications and actively collaborates on international research projects involving satellite observations and numerical modeling of ocean processes.

Gustavo A. Ovando-Montejo is an Assistant Professor at Utah State University (USU), based at the Blanding campus, where he teaches in the fields of environmental studies and geography. His expertise includes Geographic Information Systems (GIS) and research on human-environmental systems, with a particular focus on land use issues affecting Indigenous rights and ecosystem services. Ovando-Montejo is recognized for implementing innovative teaching strategies, especially for online courses, and is dedicated to mentoring Native American students and supporting outreach opportunities for local Tribes. In 2024, he was named the QCNR (Quinney College of Natural Resources) Teacher of the Year for his engaging and dedicated teaching across various classroom settings.

Peter Lindstrom is a computer scientist in the Center for Applied Scientific Computing at Lawrence Livermore National Laboratory. His research focuses on data compression, scientific visualization, and high-performance computing. Peter earned a Ph.D. in Computer Science from Georgia Institute of Technology in 2000 and holds B.S. degrees in Computer Science, Mathematics, and Physics from Elon University. He is the chief architect of the $R\&D$ 100 award-winning ZFP compressor, and is an IEEE Computer Society Distinguished Contributor.

Valerio Pascucci is the founding director with the Center for Extreme Data Management Analysis and Visualization (CEDMAV), University of Utah. He is also a faculty with the Scientific Computing and Imaging Institute, a professor with the School of Computing, University of Utah, and a laboratory fellow, of PNNL. Before joining the University of Utah, he was the data analysis group leader of the Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, and adjunct professor of computer science with the University of California Davis. His research interests include Big Data management and analytics, progressive multi-resolution techniques in scientific visualization, discrete topology, geometric compression, computer graphics, computational geometry, geometric programming, and solid modeling.