This article has been accepted for publication in Computing in Science & Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCSE.2024.3402389

IEEE CiSE Special Issue – Converged Computing: A Best-of-Both Worlds of HPC and Cloud

Jetstream2: Research Clouds as a Convergence Accelerator

David Y. Hancock, *Indiana University, Bloomington, IN, 47408, USA*
Jeremy Fischer, *Indiana University, Bloomington, IN, 47408, USA*
John Michael Lowe, *Indiana University, Bloomington, IN, 47408, USA*
Scott Michael, *Indiana University, Bloomington, IN, 47408, USA*
Le Mai Weakley, *Indiana University, Bloomington, IN, 47408, USA*

Abstract—Over the past decade, the convergence of Cloud and High-Performance Computing (HPC) has undergone significant movement. We explore the evolution, motivations, and practicalities of establishing on-premise research cloud infrastructure and the complementary nature with HPC and commercial resources; under the belief that research clouds serve a unique role within research and education as a convergence accelerator. This role is highlighted through exploring the design trade-offs in architecting research clouds versus HPC resources, focusing on the balance between utility, availability, and hardware utilization. The discussion provides insights from experiences with the National Science Foundation-supported Jetstream and Jetstream2 systems, showcasing convergence technologies and challenges. A variety of real-world use cases are provided that show the interplay between these computing paradigms; exploring use in research and education for interactive and iterative development, as an on-ramp to large-scale resources, as a powerful tool for education and workforce development, and for domain specific science gateways.

Cloud and high performance computing (HPC) have been converging in a dynamic, bi-directional manner for more than a decade. Here we discuss some of the convergence technologies utilized, our motivations for creating and sustaining on-premise research cloud infrastructure, and the usage examples that highlight the fluidity between cloud and HPC environments. This convergence has been an ongoing process, with early testbeds such as the deployment of Magellan within the Department of Energy (DOE) in 2009 [1] and OpenStack production clouds deployed at large scale by institutions such as the European Council for Nuclear Research (CERN) in 2013. In the early years of cloud computing, commercial providers largely avoided HPC specific technologies and adaptations; there were exceptions within niche providers such as Penguin Computing. As provider interest grew to focus on serving a larger portion of research computational needs, in addition to hosting bare metal HPC systems, commercial cloud providers, specifically hyperscaler ones, began delivering native HPC software technologies such as Lustre, Spectrum Scale (GPFS), and Slurm, as well as hardware technologies such as InfiniBand, and a wide array of co-processors/accelerators. We set aside the hosting of standalone, bare metal supercomputers within commercial facilities as more of a operating model and contracting change than a technology convergence in this discussion, but increasingly we do see participation and interest from these providers in the community of research and education (R&E) practitioners that share deployment and operations methods.

In our own experiences architecting, deploying, and operating research cloud and HPC environments there are a series of trade-offs with respect to the ideally supported use cases. We discuss a selection of those use cases from the research cloud perspective as it relates

XXXX-XXX © 2024 IEEE
Digital Object Identifier 10.1109/XXX.0000.0000000

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in Computing in Science & Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCSE.2024.3402389

to the National Science Foundation (NSF) supported Jetstream and Jetstream2 [2] systems, deployed in 2015 and 2020 respectively, these were inspired by a modest gateway hosting environment at Indiana University (IU). We will also discuss the convergence technologies, particularly those that have a high degree of crossover from clouds to HPC systems and vice versa. First, we will expand more on these motivating factors followed by a discussion on technologies involved in this convergence, and specifically their use through a selection of science and education use cases.

MOTIVATION

Throughout the period of convergence between cloud and HPC, we have seen a steady increase in the use of cloud computing in research and education. Many national research centers, national labs, and research universities now provide cloud computing capabilities to their constituents either through efforts like the on-premise systems mentioned above or via managed access to commercial clouds. In general, this increase in cloud computing capacity has supported new or expanded research programs; cloud technologies have been used in ways distinct from HPC systems to enable research and education. Although there may have been a concern about HPC workloads moving to the cloud wholesale, there are relatively few workloads that have been wholly re-architected to be cloud native. Instead, cloud technologies have arisen to provide an HPC-like environment in a cloud.

This convergence without HPC resources being consumed by hyperscale providers is due to a series of design trade-offs in our view. The most prevalent ones we see come in the form of usability (convenience of access) versus optimal hardware utilization; and high availability versus raw performance. Specifically, the on-demand nature of cloud resources is appealing to many of the use cases discussed, and supporting that on demand capacity requires lower utilization targets than batch scheduled HPC systems. Similarly, raw HPC performance is exchanged for the use of hypervisors and virtual networking technologies that support the live migration of resources, allowing for long running workflows and rolling maintenance without scheduled downtime. These design trade-offs fuel the convergence technologies and adoption of a particular platform beyond generational familiarity and educational experience. While embedded in those trade-offs are also cost differentials, we focus on the convergence in terms of functionality for this discussion along with the ongoing challenges.

At IU, we have operated the aforementioned Jetstream and Jetstream2 cloud systems as well as a number of traditional HPC systems. Over the years, we have seen technologies, operational models, and workflow and programming paradigms arise that aim to leverage the best of both cloud and HPC while attempting to be applicable in both environments. In this paper, we give an overview of what we mean by cloud vs. HPC technology and highlight a number of technologies that are in the area of convergence and can be applied in both environments. We delve into a number of observed use cases that we have supported, and attempt to highlight the cases where there is a significant crossover between HPC and cloud technologies being employed. This can happen at a number of levels, whether it is in the hardware and service level components, the operational paradigms, the end user environments, or the overall workflows. In many cases, we see a great synergy between cloud and HPC approaches, one such example being science gateways. In other cases, we note where technologies generally considered to be HPC are being used in a cloud environment and vice versa. We also note where cloud technologies have enabled new modes of interaction with researchers and educators.

In the final sections of this paper, we survey the current landscape of converged cloud and HPC computing in research and education and highlight some of the current and future challenges. Some of the major challenges include the potential side effects of technology lock-in and the need to train and retain a workforce that can support research and education in both cloud and HPC environments. Through highlighting these challenges, converged technologies, and use cases we hope to give an overall sense of the state of cloud and HPC and show how these two approaches are coming together to support advances and innovation in research and education.

TECHNOLOGICAL CONVERGENCE

Cloud computing techniques are typically characterized by API driven instantiation of à la carte infrastructure and services on demand without time based scheduling. RESTful style HTTP transport of the API calls is common with the intent of enabling multiple methods of access, including programmatic.

One of the most archetypical and earliest cloud computing technology is object storage. OpenStack Swift, Amazon Web Services S3, and MinIO are popular implementations of object storage where objects are stored in a relatively flat two level namespace, initially using eventual consistency (with MinIO being designed with and S3 gaining strong consistency in

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in Computing in Science & Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCSE.2024.3402389

December 2020), and either shared secrets or anonymous identity and access control. Relaxing the consistency of POSIX semantics reduces the amount of synchronization needed to implement object style storage making it more suitable than parallel filesystems used in HPC for the read intensive data sets involving large numbers of small files found in biology and AI. An S3 adaptor for HDF5, a file format commonly used in geosciences, has been developed by the HDF Group to transparently bring object storage to applications without code changes. We also see more storage products that either provide S3-style interfaces or use objects natively, that provide both POSIX and object interfaces in use by HPC centers today. From the commercial cloud perspective, it is also notable that the three largest cloud providers (as well as others) all offer some form of managed Lustre storage within their service offerings.

The storage of operating system images is metered in commercial clouds and often managed with an allocations process and quotas in on-premise clouds. Coupled with the effort required to apply security and bug fixes to images, this incentivizes using images created and maintained by upstream providers. To affect a customized image deployment to bare metal or virtual machines, configuration management such as Ansible, SaltStack, Chef, or Puppet is applied immediately after the instantiation of a generic base image. HPC centers now commonly use these configuration management techniques as a way to ensure high quality consistent deployment of ancillary services with minimal effort. As a service to users, the Jetstream2 project provides a set of images and Ansible scripts sufficient to deploy an OpenHPC-based virtual cluster with Slurm that dynamically scales, deleting and creating compute instances based on the number of jobs queued. This functionality is made possible by repurposing the Slurm power management hooks for powering off idle HPC compute nodes, with scripts for creating and deleting OpenStack instances and hence is an example of composite HPC and cloud.

Another area of convergence is software and environment management. Applications in HPC centers are usually delivered via a shared filesystem and users manage their own environment by augmenting a base system image through Lmod or a similar flexible environment management system. In a cloud environment users can create customized virtual machine (VM) images, or run containers in a vanilla VM image. While containers have been around in some form for a number of years, many researchers began adopting their use with the popularization of Docker. Containers allow users to easily move between a variety of compute resources, including workstations, cloud, and HPC. With the advent of technology like Singularity and Podman that allow users to execute containers without privileged access, the uptake of containers on HPC has increased significantly as seen by the number of papers and workshops focused in this area at HPC-specific events.

Conversely, Jetstream2 has adopted a more traditional HPC method for the delivery of software as a way to reduce the number, and complexity of, maintained base images, while maximizing flexibility in user environments and minimizing the amount of staffing resources required to maintain images. Software is installed in a centralized repository that is auto-mounted by users' VMs. Users can then access those software installations via Lmod. This allows base images to be updated through an automated CI/CD pipeline to pull the latest updates into those images on a regular basis. Other research infrastructure projects have taken a different approach from mounted filesystem shares. The CernVM Filesystem (CVMFS) provides a POSIX-compliant, FUSE-mounted file system delivered via standard web servers. This approach allows for user-space mounting and also allows for a simple replication of file systems to make it possible to deliver a consistent application or data store to geographically dispersed research infrastructure. We see this class of converged technology with Galaxy [4] and several Open Science Grid [11] consortium projects using CVMFS optimizing data and application delivery through a network of multi-stratum storage servers.

The Jetstream2 roadmap was crafted for pragmatism in the midst of this convergence, under the constraints of the award, and on prior experience with Jetstream. The team realized from prior work with researchers that cloud-native options were often a better fit for commercial cloud services rather than the bespoke uses in research workflows. Utilizing an approach from HPC with Lmod and a shared application filesystem allowed researchers to more readily leverage a cloud resource because it gave them a somewhat familiar environment. Further, primarily using POSIX-compliant block storage for additional user-defined storage, rather than a cloud-native object store environment, met the needs of domain science researchers whose software has not yet adopted an object store model. While object storage is available in Jetstream2, it's generally the third tier of storage for most individuals. The team opted not to provision a shared high performance parallel filesystem for simplicity, noting that the usage models of most cloud researchers on the previous system did not require the speed (or complexity) in running a VM-accessible

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in Computing in Science & Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCSE.2024.3402389

Lustre filesystem. Instead, making project controlled NFS sharing easier to manage was oft-requested functionality, accomplished using CephFS and Manilla on Jetstream2.

Jetstream2 has contributed to convergence at other academic institutions, expanding from the dual-site model of the first Jetstream project. Jetstream2 has a regional cloud model that allows OpenStack resources to be folded into the NSF CI ecosystem while still maintaining local control. This regional cloud aspect has benefits that may not seem obvious. Commercial clouds have provided geographically dispersed resources for some time, but demonstrating a federated model in academia with the ability to have independent organizations operating the environment in unison has been limited. The regional cloud model serves to show that with minimal coordination, federated clouds can be deployed relatively easily amongst different institutions and permit an organization to maintain a level of control, as well as data locality. It also contributes to the cloud administration expertise within host institutions, while simultaneously introducing research-centric cloud services to organizations that may not have had an on-premise cloud environment. While additional regional clouds are not widespread, multiple organizations have worked with us to provision discrete systems and are considering federation, clearly recognizing the need of research cloud services to supplement and enhance HPC services.

USE CASES

Through the operation of Jetstream and Jetstream2 we have seen cloud technologies used in a number of different ways to address research and education challenges. The use cases below highlight some areas where cloud technologies have provided new solutions to old problems in the HPC space, where clouds and HPC have complemented each other to enhance the capabilities of both, and also where new converged technologies have been deployed to minimize friction in using both environments and maximize the reuse of code, technologies, and operational paradigms. Here we highlight use cases from interactive usage, to iterative development and scaling to HPC, to education, to domain science gateways. Jetstream2 has provided variety of avenues that bring the cloud to HPC/HTC (high throughput computing) and vice versa. These use cases highlight some of the paths researchers may take between these two computing landscapes.

Interactive Computing

Many disciplines have traditionally used large-scale, publicly available research cyberinfrastructure to conduct experiments and investigations that follow a typical pattern of simulation and modeling. This workflow often includes an investigation that requires a small number of simulations, in some cases only one very large simulation, where the input data for the simulations are small to modest-sized and the output data have a relatively large data volume. These output data then require subsequent processing and analysis to determine the results of the study. Sometimes this workflow paradigm is expanded to run ensembles of simulations designed to investigate a large volume of the input parameter space; the resulting ensemble of simulation outputs can then be intercompared to find areas of interest in the input parameter space for further investigation. These types of workflows are prevalent in a number of research areas powered by simulation, including climate and weather modeling, computational fluid dynamics, simulation of materials properties, molecular dynamics, and astrophysics. The aforementioned workflows typically map very cleanly onto the resources provided by batch-scheduled HPC systems.

In recent years, the ubiquity and availability of large-scale data sets and the ability to collect large volumes of data from a variety of instruments have led to investigations that employ workflows that are both computationally intensive and also have large input data sets. For these workflows it is quite often the case that some experimentation with the analysis using a smaller sub-sample of the data is required to determine the best approach for the investigation. This same approach is sometimes used in the downstream processing of the data output from simulation workflows. This usually requires some amount of interaction with the data set to evaluate the data quality and determine the best approach, followed by some interactive experimentation and iterative refinement before the analysis steps are finalized and can be applied to the full, larger scale data set. A number of statistical methods use this approach along with many machine learning and deep learning methods. For example, the refinement of hyperparameters in the training of a neural network, or determining the best parameters to use in a clustering algorithm. Another example is interactively exploring the visualization of outputs from a large scale simulation.

In contrast to simulation and modeling workflows, these workflows do not map well onto the resources provided by batch-scheduled HPC systems for a number of reasons. In many cases, the compute resources

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in Computing in Science & Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCSE.2024.3402389

in an HPC system are not accessible to the public network, but are put behind login nodes and accessed through these nodes. This introduces a need to tunnel through the login node to access the compute resources. There have been solutions that address this problem (e.g. OpenOnDemand [9]), but this is still a workaround. The other issue, which tools like OpenOnDemand still struggle to address, is that interactive computing is difficult to schedule by its very nature. Generally, this is addressed by users allocating resources for as long as they think the interactive session will take, but this is often inaccurate and produces less-than-ideal scheduling outcomes. Moreover, the end user expectation for an interactive resource is little to no wait time to access that resource. This is built into the cloud model but is not the case with most batch-scheduled HPC systems. The underlying assumption of batch-scheduled HPC systems is an abundance of defer-able work of fixed duration and a scarcity of hardware, while cloud assumes a scarcity of real or invented currency and work of undefined duration. Scarcity of hardware is considered a system fault in cloud computing, requiring the user to retry. There are nascent efforts to mix these two fundamental world views such as CoreWeave's SUNK, and Slurm scheduled Kubernetes, being released in early 2024. Neither model is universally suited for all use cases and hybridization is not mature enough to declare it an unconditional winner.

Research Clouds as an On-ramp to Larger HPC Systems

It is a common use case for researchers on Jetstream2 to use the cloud resource interactively as a larger workstation environment using virtual desktops. This functionality often includes common scientific desktop applications that may or may not scale well to traditional HPC, batch system computing, or just are outside the researcher's comfort zone to use in that manner. This can take a number of forms, such as the above mentioned utilization of visualization and analysis tools to analyze the output of HPC workflows, to researchers moving workflows that had previously lived entirely on their personal or lab workstation to another platform. This latter use case is often driven by the computational or data storage needs having exceeded what the local resources can comfortably provide in terms of memory or storage, but the research team is inexperienced with HPC or does not think HPC resources are warranted. In some cases, these workflows fit nicely into a single or few VMs in a cloud environment, and researchers will use the cloud environment to remain productive. In other cases, the research team may continue to scale up their computational and data storage needs and then find their way onto HPC/HTC resources.

Another key use case for clouds as an on-ramp to HPC is in prototyping and development. Cloud environments can provide isolated, consistent environments with more resources than a software engineer might typically have. Furthermore, for creating smaller scale, proof-of-concept workflows, research clouds allow testing before scaling up to HPC or commercial cloud when more resources are needed. There are two areas of HPC and cloud convergence that make this use case particularly interesting. First is the use of containers in both cloud and HPC environments. Containerization allows developers to package an environment that is nearly identical to the environment on a large scale HPC system and run it almost anywhere. This allows developers to take their workflow development onto a cloud resource and minimizes the overhead when migrating to the HPC systems. Some operators of large scale HPC systems such as the National Energy Research Scientific Computing Center (NERSC) are even looking to completely containerize their HPC platforms [8], which would make the transition from a cloud development platform to running on HPC even more straightforward. The second area of convergence is the ability to easily deploy a batch scheduled cluster inside a cloud environment, described previously as a virtual cluster. This capability, coupled with containerized environments, allows developers to deploy miniature versions of much larger scale HPC resources and test their workflows end-to-end in their "native" environment.

Use Case: Event Horizon and M87 black hole. The Event Horizon Telescope team demonstrated a use case of prototyping workflows on research clouds before scaling up their work to other resources. In 2019, a team of researchers from around the world shared the first images of a black hole at the center of the Messier 87 galaxy. These images were generated from workflows created on Jetstream. Based on the success of the prototyping, they were able to scale up to commercial cloud and HPC resources for their full analysis runs processing petabytes of data. Dr. Chi-kwan Chan, associate astronomer at University of Arizona who leads the EHT Computations and Software Working Group, said “The production run was actually carried out on Google Cloud, but much of the early development was on Jetstream. Without Jetstream, it is unclear that we would have a cloud-based pipeline at all.” [3] This work leveraged a research cloud to de-

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in Computing in Science & Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCSE.2024.3402389

velop the workflows that led to a remarkable scientific achievement and also made use of the commercial cloud for HPC. This use case presents converged computing in its most direct form and may be a model to others looking to create workflows that deal with large-scale inputs.

Use Case: Oak Ridge National Lab prototyping for leadership class HPC. Leadership class HPC machines often have strict guidelines for usage. These may include proving that a researcher's application can scale to leverage the resource as intended, solving problems that can only take advantage of the largest available HPC systems. Oak Ridge National Lab (ORNL) solved this problem for several researchers using more readily available cloud resources to demonstrate that researchers' code and computation plans were ready for use on the Summit supercomputer.

ORNL maintains its own in-house cloud system for similar code development and experimentation, but due to demand, it was often completely utilized and not available for researchers to experiment with scaling up their code. The ORNL lead cloud engineer worked with the Jetstream cloud operators to do testing on the Jetstream cloud resource that was located at the Texas Advanced Computing Center (TACC). These test runs spanned several allocations with multiple PIs and utilized virtual Slurm clusters or VMs managed with the OpenStack Software Development Kit (SDK), cloud-init scripts, and utilizing a message passing interface. These test runs utilized around 1,000 cores generally using 24-core VMs.

These tests have tapered off for a variety of reasons. There may be future opportunities for this sort of scaling testing/prototyping with ORNL and other institutions for HPC workflows in the cloud. Whether or not future collaborations materialize, this convergence case with a need for on-demand, on premise cloud resources to prepare for using one of the largest HPC systems in production at that time is an example worth noting.

Research Clouds for Training and Workforce Development

It is without question that the state of science today requires advanced computing techniques and resources, but while research necessitates such tools, not all institutes and scientists have access to systems that will allow them to scale their work and push on the boundaries of research. Without providing a means to use large-scale cyberinfrastructure, the gap between communities who are able to advance their work and those who cannot, widens considerably, simply based on what is made available to them without a significant monetary burden. Not only can research clouds bridge this gap by providing access to these cutting edge technologies, the ability to deploy consistent, reproducible, environments allows for training on advanced computing techniques and infrastructure, thereby creating sustainability in the state of the practice and fostering inclusivity within the landscape of science. Certainly, with the growing need and popularity of AI workflows, we have also found that the ability to leverage research clouds to provide virtual GPU resources (vGPUs), by partitioning physical accelerators, has allowed a larger community to access technology that is in high demand, and often scarcely available, for education and workforce development.

Use Case: Unidata Science Gateway for Education. Unidata is one of the University Corporation for Atmospheric Research (UCAR)'s Community Programs (UCP) and has a mission to serve researchers and educators in Earth systems science. As part of this mission, they have created a science gateway for training both future and current researchers.

The Unidata Science Gateway is an auto-scaling JupyterHub, built from the Zero To JupyterHub repository utilizing Kubernetes hosted on the Jetstream2 cloud. It provides a means to analyze and visualize Earth system data in classrooms and workshops and also provides real time data via other Unidata services for research analysis. Users receive their own prepared workspace and dedicated storage via the JupyterHub and the Kubernetes-backed construction which allows Unidata to scale up or down as necessary for courses and workshops.

While the gateway service began prior to the COVID-19 pandemic, a large spike in growth happened when students and researchers went remote. The Unidata gateway has supported over 1,000 students through semester long courses and workshops, taking the charge of preparing the next generation of researchers in a structured, scalable and resilient way [7]. Even with the return to in-person conferences and higher education, the use of the Unidata Science Gateway keeps increasing, prompting the Unidata team to continually work on improving the system for the best user experience they can provide. In addition to the student education aspects, Unidata is creating and presenting workshops on topics like MetPy using their gateway, helping researchers update their workflows with modern Python libraries and demonstrating techniques for plotting satellite, model forecast data, and surface observation data as well as other research top-

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in Computing in Science & Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCSE.2024.3402389

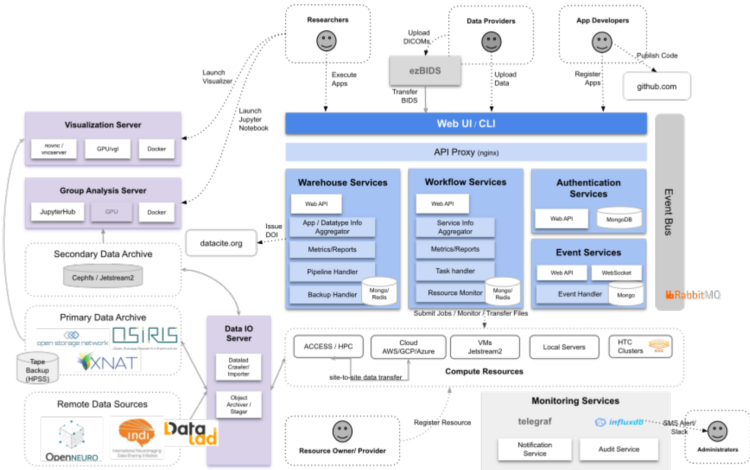

FIGURE 1. Brainlife Service Relationship Diagram. Image courtesy of Franco Pestilli.

ics like remote and local data access and management using their workflows.

Use Case: Infrastructure for Accessible Environmental Data Science Education. Environmental Data Science (EDS) is a data-intensive subdiscipline that cuts across a wide gamut of research fields, from environmental and computational sciences to biological and social sciences. Necessarily this would also imply that the computing and data science skills of the researchers in EDS would also span a wide gamut and a natural question would arise of how does one train such a cohort in this field in order to familiarize themselves with tools and techniques in advanced computing, to allow them to translate or scale their research to other platforms, be it cloud or HPC.

In February 2023, the EDS working group within the University of Colorado (CU) Boulder Earth Lab and the Environmental Data Science Innovation and Inclusion Lab (ESIIL) deployed a reproducible and scalable cyberinfrastructure on Jetstream2 via the Cloud Automation and Continuous Analysis Orchestration (CACAO) UI, enabling a group scientists with varying skills in computing and data science to leverage 40,000 hours of compute time over a 4-day workshop, creating an open framework that can be replicated for educational purposes in any data- or compute-intensive discipline [6]. Through the CACAO interface, the working group was able to deploy a JupyterHub server orchestrated by Kubernetes on Jetstream2, without needing to setup Kubernetes themselves. By reducing the time needed to set up these environments, the workshop attendees were able to not only absorb information about the infrastructure they were working on, but also interact with the resource immediately, allowing them to focus on learning data science skills, and in a tangible way that demonstrated how access to these resources could impact their research. This in turn empowers researchers to seek out resources and tools to scale their work, steering them to the path of advanced computing, be it on cloud or HPC resources, converging these two paths together.

Domain Specific Science Gateways

The convergence of HPC and cloud computing occurs even more naturally within science gateways. Through lower-barrier web-based portals, science gateways have been central to making advanced research infrastructure and suites of domain specific applications available to communities of scientists and educators without having individuals each go through the process of requesting access and submitting jobs, or be limited by the constraints of job wall time policies and regular system maintenance downtime. By providing a platform to host multiple types of browser-based services, either as portals to execute scientific workflows directly or as a gateway to orchestrate computation on other HPC and cloud compute resources, Jetstream2 has become a home for several science gateways seeking to broaden access to compute resources to their community and scale their operations to reach a wider audience. We outline two such cases below.

Use Case: Brainlife. Brainlife [5] is a science gateway for neuroscience analysis. The project, led by Franco Pestilli and his lab at University of Texas Austin, is a web-based platform that allows people to analyze MRI, EEG and MEG data on a variety of HPC and

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in Computing in Science & Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCSE.2024.3402389

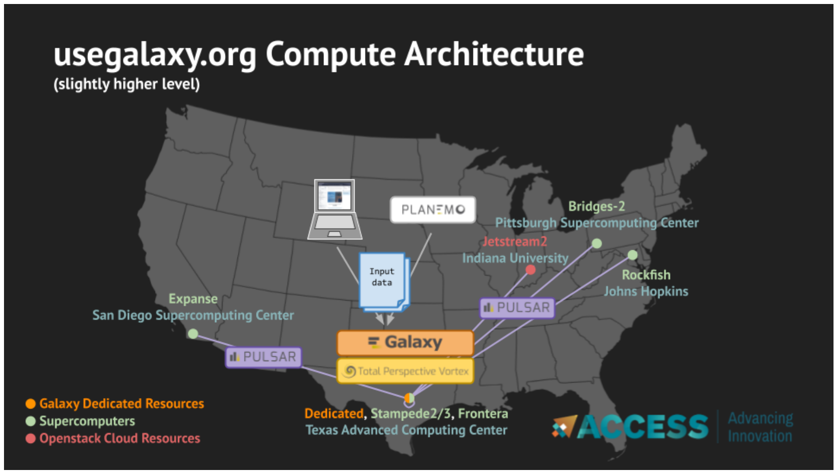

FIGURE 2. UseGalaxy.org Compute Architecture.

cloud resources, including Jetstream2. Figure 1 shows the system architecture diagram for the Brainlife platform. As detailed in the diagram, Brainlife consists of several microservices that are distributed across multiple Docker hosts and various VMs on Jetstream2. The microservices running on Jetstream2 include the web UI, event and authentication services, as well as the orchestration of the jobs on the HPC and cloud resources (both commercial cloud and Jetstream2) on which the workflows and applications run. In addition to providing a default set of both HPC and cloud resources, Brainlife also allows users to register compute resources, whether they are clusters at an institution or a cloud resource, and make them available to a user or set of users, thereby bringing these two compute paradigms closer to one ecosystem. By simply signing up on this platform, neuroscientists gain the ability to run applications necessary for their research on both HPC and cloud resources without the lead-up time necessary for requesting access, installing software or being subject to scheduler constraints that come with traditional HPC. In this capacity, Jetstream2 is not only providing the venue for convergence by hosting the gateway, it is also a participant in the convergence in terms of belonging to the pool of compute resources that power the analyses carried out on the Brainlife platform.

Use Case: The Galaxy Project. Galaxy [4] is a mature web-based platform initially released in 2005. It has gone through a number of iterations and updates with the intent of making computational biology workflows available to researchers that may not have extensive experience as system administrators or programmers. The platform was originally designed for biological analysis focusing on genomics but the open toolkit design has allowed for expansion into genome assembly, proteomics, SARS-CoV-2 public data, and now into machine learning with the Galaxy-ML toolkit. The initial design of the Galaxy framework started as a number of standalone tools that were hosted as web applications on dedicated servers at HPC centers. Computationally intensive calculations were dispatched to the HPC backend. These efforts began prior to the modern commercial cloud era, Amazon launched S3 cloud storage and EC2 in 2006. Over time, the Galaxy framework has expanded by adding a large number of tools and data sets along with deploying on multiple types of infrastructure.

The Galaxy Project services thousands of users per day through their science gateway that utilizes a number of compute resources. They maintain dedicated services at the TACC data center that host their web front-end and a small dedicated HPC cluster for Galaxy computation. On Jetstream2 Galaxy leverages cloud-native methods to provision HPC-like infrastructure routing workflows to auto-scaling virtual Slurm clusters as well as Kubernetes-based auto-scaling clusters and individual on-demand instances for interactive workloads. Additionally, they utilize other compute resources in the NSF's Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) national cyberinfrastructure (see Figure 2). They developed the Galaxy Total Perspective Vortex (TPV) to solve the question of where to send Galaxy jobs across their available resources. With this workflow solver, Galaxy can easily send jobs to the best resource available for the scope of the workflow. These

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in Computing in Science & Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCSE.2024.3402389

include four HPC/HTC systems in ACCESS as well as the Jetstream2 cloud. Additional resources such as the CVMFS Stratum One software servers are hosted on Jetstream2 making Galaxy a true convergence point in HPC and cloud, bringing cloud hosting and computation as part of the possible pool of computation with HPC resources to serve research needs.

BROADER ECOSYSTEM AND OPEN CHALLENGES

Although the focus of this discussion has been use cases enabled by the Jetstream and Jetstream2 cloud resources, there are a number of other cloud resources and services that have been made available throughout the NSF ecosystem over the past several years. These include more experimental cloud deployments like Chameleon Cloud [12] and Cloudlab as well as services to help research teams leverage the commercial cloud like Cloudbank. Cloud-like technologies such as virtual clusters and container runtimes have also been enabled at a number of DOE computing facilities and many NSF ACCESS resource providers with composable or cloud approaches leveraged by Purdue's Anvil, Pittsburgh's Bridges-2, and others via collaborations through projects like OSG and the National Research Platform. We continue to see advances in the convergence of technologies between cloud and HPC, from system deployment and management like HPE's Cray System Management software which is moving toward a multi-tenancy model allowing both HPC and cloud resources to exist in the same physical machine and be managed by the same management stack, recently demonstrated at the Swiss National Computing Centre; to advances like Openstack's Ironic and Blazar which bring bare-metal deployment and advanced scheduling capabilities to cloud resources.

Although much progress has been made, and there is a great deal of promise in a future that combines cloud and HPC technologies, there remain a number of technical and structural challenges in bringing cloud and HPC together. The virtualization and integration of accelerators, particularly when considering live migration, remains a challenge in Openstack and other cloud platforms. Beyond technical challenges, having a workforce that can deploy, operate, and support both cloud and HPC resources, and help individuals get the most out of these systems through hybrid approaches, is extremely challenging. Such talented staff are very difficult to find, and even harder to retain with both cloud and HPC talent being in high demand in the private sector.

DISCUSSION AND CONCLUSION

It is our belief that these use cases highlight the utility of on-premise research clouds, particularly where technology convergence allows the R&E community to couple these environments with commercial cloud and traditional HPC resources in a manner that promotes productivity towards an individual's goal, without forcing them into a pre-determined path for their compute and storage needs. While specific system design goals may not allow full convergence in the immediate future, there is a parallel visible within systems deployment of research clouds and HPC systems to early UNIX/Linux application deployment, particularly much of the GNU software suite. Those command line applications were typically purpose-built as a tool within a larger array of programs but collectively served a broad audience and were coupled together with scripts by passing standard out to standard in via pipes or redirection. In the same way we see research cloud and HPC systems being very complementary and best utilized in a coupled manner. While the convergence technologies discussed such as containers, virtual clusters, and crossover storage technologies very much aid in coupling these environments, they do not obliviate the need for each other due to core differences in scheduling paradigms, tightly-coupled versus distributed application scaling, and levels of interactivity or utilization targets. Ultimately the system or approach selected by an individual researcher or educator may be due to a variety of factors that fall well outside of the architecture choices for a resource, and based on proximity, familiarity, ease of access, or domain-specific norms. By providing research cloud environments, or strong support and facilitation for researchers operating in the commercial cloud, we allow researchers and educators to leverage the environment and approach they're most comfortable with, accelerating the adoption of technologies that are already converged, while allowing time and space for additional adaptation.

ACKNOWLEDGMENTS

The IU Pervasive Technology Institute was established with funding from the Lilly Endowment, Inc. This material is based upon work supported by the National Science Foundation under Grant No. 1445604 and 2005506. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF. We thank the following individuals for their contributions as well as their tireless support for the R&E community: Craig A. Stewart, George

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in Computing in Science & Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCSE.2024.3402389

Turner, Nirav Merchant, Matthew Vaughn, Therese Miller, Winona Snapp-Childs, Edwin Skidmore, Gwen Jacobs, Jane Combs, Chris Martin, Julian Pistorius, Eric Coulter, Marlon Pierce, and Suresh Marru.

REFERENCES

1. Lavanya Ramakrishnan, Piotr T. Zbiegel, Scott Campbell, Rick Bradshaw, Richard Shane Canon, Susan Coghlan, Iwona Sakrejda, Narayan Desai, Tina Declerck, and Anping Liu. 2011. Magellan: experiences from a science cloud. In Proceedings of the 2nd international workshop on Scientific cloud computing (ScienceCloud '11). Association for Computing Machinery, New York, NY, USA, 49–58. https://doi.org/10.1145/1996109.1996119
2. David Y. Hancock, Jeremy Fischer, John Michael Lowe, Winona Snapp-Childs, Marlon Pierce, Suresh Marru, J. Eric Coulter, Matthew Vaughn, Brian Beck, Nirav Merchant, Edwin Skidmore, and Gwen Jacobs. 2021. Jetstream2: Accelerating cloud computing via Jetstream. In Practice and Experience in Advanced Research Computing (PEARC '21). Association for Computing Machinery, New York, NY, USA, Article 11, 1–8. https://doi.org/10.1145/3437359.3465565
3. TACC Supercomputers Play Pivotal Role in Event Horizon Telescope's First-Ever Black Hole Image. [Online]. Available: https://www.hpcwire.com/off-the-wire/tacc-supercomputers-play-pivotal-role-in-event-horizon-telescopes-first-ever-black-hole-image/ (URL)
4. The Galaxy Community, The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2022 update, Nucleic Acids Research, Volume 50, Issue W1, 5 July 2022, Pages W345–W351, https://doi.org/10.1093/nar/gkac247
5. Pestilli, Franco, et al. Brainlife.Io: A Decentralized and Open Source Cloud Platform to Support Neuroscience Research. arXiv:2306.02183, arXiv, 11 Aug. 2023. arXiv.org, http://arxiv.org/abs/2306.02183.
6. McIntosh, Tyler L, et al, "Cyberinfrastructure deployments on public research clouds enable accessible Environmental Data Science education," *Practice and Experience in Advanced Research Computing (PEARC '23)*, 2023, pp. 367–373. (Conference proceedings)
7. Chastang, Julien; Espinoza, Ana; Ramamurthy, Mohan (2023). Unidata Science Gateway: Past, Present, and Future Plans. figshare. Presentation. https://doi.org/10.6084/m9.figshare.22043642.v1
8. D. Fulton, L. Stephey, R. Canon, B. Cook, and A. Lavely, "Containers Everywhere: Towards a FullyContainerized HPC Platform," *Cray User Group (CUG '23)*, 2023. (Conference Proccedings)
9. D. Hudak, D. Johnson, A. Chalker, J. Nicklas, E. Franz, T. Dockendorf, and B. L. McMichael, "Open OnDemand: A web-based client portal for HPC centers," *Journal of Open Source Software.*, vol. 3, no. 25, pp. 622, 2018 (Journal)
10. CernVM File System. [Online]. Available: https://cernvm.cern.ch/fs/ (URL)
11. Pordes, R., Petravick, D., Kramer, B., Olson, D., Livny, M., Roy, A., Avery, P., Blackburn, K., Wenaus, T., Würthwein, F., Foster, I., Gardner, R., Wilde, M., Blatecky, A., McGee, J., & Quick, R. (2007). The open science grid. J. Phys. Conf. Ser., 78, 012057. https://doi.org/10.1088/1742-6596/78/1/012057
12. K. Keahey, J. Anderson, Z. Zhen, P. Riteau, P. Ruth, D. Stanzione, M. Cevik, J. Colleran, H. S. Gunawi, C. Hammock, J. Mambretti, A. Barnes, F. Halbach, A. Rocha, and J. Stubbs, "Lessons Learned from the Chameleon Testbed," in *Proceedings of the 2020 USENIX Annual Technical Conference (USENIX ATC '20)*, 2020

David Y. Hancock is the director for advanced cyberinfrastructure in IU's Research Technologies division, a center within the Pervasive Technology Institute, and part of University Information Technology Services. He is responsible for directing local and national HPC, storage, and cloud resources for research. Hancock is the PI for the Jetstream2 project. He is also Co-PI on the NSF Cyberinfrastructure Professionals Innovating and Broadening the Adoption of Advanced Technologies award (CI PIVOT), and part of the NSF ACCESS Operations project. Hancock has been an IEEE and IEEE Computer Society member for over 10 years and an active member in leadership of multiple research computing organizations. Contact him at dyhancoc@iu.edu.

Jeremy Fischer Jeremy Fischer leads the Research Cloud Infrastructure team for IU's Research Technologies division. He oversees operations for multiple research clouds, including the Jetstream2 system, of which he is a co-investigator. He worked on the first Jetstream cloud as the education, outreach, and training lead, XSEDE compatible clusters projects and has served as a Unix system administrator for infrastructure services such as system-wide mail, DNS, and DHCP prior to working in research services.

John Michael Lowe is a senior cloud engineer at Indiana University, Bloomington, Indiana. Contact him at jomlowe@iu.edu.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in Computing in Science & Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCSE.2024.3402389

Scott Michael is the director of the Research Software and Solutions group in IU's Research Technologies. His current research interests include performance analysis and optimization, resource utilization and efficiency of research computing centers, and standardizing benchmarking methodologies for emerging software areas like deep learning frameworks. Michael received his Ph.D. degree in astrophysics from Indiana University. Contact him at scamicha@iu.edu

Le Mai Weakley leads the Research Cloud Services team within IU's Research Technologies. Her responsibilities include leading the education, outreach and training efforts for Jetstream2. Weakley received her Ph.D. degree in Mathematics from Indiana University. Contact her at llnguyen@iu.edu.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/