The "Geddes" Composable Platform - An Evolution of Community Clusters for a Composable World

Preston M. Smith
Research Computing
Purdue University
West Lafayette, Indiana
psmith@purdue.edu

Erik Gough
Research Computing
Purdue University
West Lafayette, Indiana
goughes@purdue.edu

Alexander Younts
Research Computing
Purdue University
West Lafayette, Indiana
ay@purdue.edu

Brian Werts
Research Computing
Purdue University
West Lafayette, Indiana
bwerts@purdue.edu

Thomas J. Hacker
Dept. of Computer and Info. Technology
Purdue University
West Lafayette, Indiana
tjhacker@purdue.edu

Norbert Neumeister
Department of Physics and Astronomy
Purdue University
West Lafayette, Indiana
neumeist@purdue.edu

Jennifer Wisecaver
Department of Biochemistry
Purdue University
West Lafayette, Indiana
jwisecav@purdue.edu

***Abstract*—New usage patterns of computing for research have emerged that rely on the availability of flexible, elastic, and highly specialized services, that may not be well suited to traditional batch HPC. A new approach that updates and evolves the research computing ecosystem is needed to respond to these needs. This new model, a Kubernetes-based "Community Composable Platform", builds upon Purdue's Community Cluster program to provide cost effective, highly responsive, and customizable composable computing solutions for domain science and education in a variety of communities.**

***Index Terms*—containers, kubernetes, rancher, composable, cloud**

I. Introduction

The scientific community powered a revolution in supercomputing by leveraging high performance, low cost commodity hardware and open source software to build "Beowulf" [1] clusters - an architecture that is now ubiquitous. In 2004, Purdue University established the Community Cluster Program [2] with a sustainable model of incentives, reliable facilities, and highly skilled central support for "Beowulf" style HPC clusters. The Community Cluster Program incentivized researchers to collectively contribute to building a centralized campus high performance computing (HPC) community in place of creating isolated laboratory level clusters. The Purdue program is highly successful, with 210 faculty active in the program and those faculty earning over $289 million (56% of Purdue's total) in research awards in fiscal year 2020.

Although the cluster program has been successful, capability gaps remain in the campus research computing ecosystem. Growing domain science demand for interactive [3] machine learning and data analytics not reliant on batch HPC often requires additional supporting services such as databases, web servers, file shares, scientific notebook systems, and other custom cyberinfrastructure services, delivered in such a way to support heterogeneous workflows requiring both services and batch computing.

Composability is one such method of delivering these capabilities. "Composable infrastructure treats compute, storage, and network devices as pools of resources that can be provisioned as needed, depending on what different workloads require for optimum performance." [4]

The campus research computing ecosystem must evolve beyond batch to incorporate composable and elastic cloud infrastructure that can be customized for the unique needs of diverse domain science communities.

This material is based upon work supported by the National Science Foundation under Grant No. 2018926.

II. Current Environment and Gaps

The core of Purdue's cyberinfrastructure strategy, Purdue has operated a world-class Community Cluster Program [2] since 2004 - each year deploying an HPC system on the order of 550 nodes, with approximately 70 faculty groups investing in each system. In total, over 210 faculty groups from 60 academic departments and every academic college invest funds in the program, with over 1200 active users each year.

Information Technology at Purdue (ITaP) and the faculty share the capital costs of the program, with faculty purchasing compute nodes with grant or institutional funds; and ITaP centrally funding shared network and storage infrastructure. Professional administrative and computational science staff are centrally funded.

The Community Cluster Program is currently built around batch HPC architectures, with job queues, high-speed interconnects, and parallel filesystems. These architectures are well suited to broad sets of usage patterns in modeling, simulation, and data analysis.

However, increasingly, new demands from researchers reinforce that batch HPC is not as ideal for many emergent usage patterns such as on-demand or interactive use, custom or elastic software stacks, web applications, database servers, or dynamic, heterogeneous workflows.

Cloud computing techniques provide a wealth of new capabilities and components that, like HPC clusters, could be exploited and provided institutionally for the research community. Researchers are using virtualization and containers today, and exploring the use of public and private clouds, but experience reveals gaps.

- First, public cloud cost is prohibitive, and fails to leverage existing infrastructure investments already made by academic institutions [5] [6] [7] [8].
- Second, academic research groups rarely have the financial means to truly benefit from the public cloud's ability to burst for short periods of time. Capital investments that produce more total resources, even if over a long period, are usually preferred.
- Third, unlike servers deployed by an IT staff, researchers moving to the cloud must run their own security operations. The cost of not configuring services correctly has often been highly visible leaks of data.
- Fourth, as with traditional clusters, the skill and effort level required to install, operate, secure, support, and manage private cloud systems is tremendous. It is often beyond the skill level of graduate students or individual researchers within a laboratory, and requires dedicated expert staff leveraged at institutional scale.

III. Goals

A. Define a New Approach

Significant collective amounts of computing capabilities exist today across campuses. Tools to fully harness and exploit this power for research use are limited to cycle harvesting systems used for high throughput computing [9]. Increasingly, systems in individual labs take advantage of what is now commonplace virtualization solutions, or container technology such as Docker [10]. Virtualization and containers provide exciting capabilities for researchers with reproducible, publishable and shareable infrastructure artifacts.

The NSF "CI2030: Future Advanced Cyberinfrastructure" report [11] describes "An integrated cyberinfrastructure that reaches from university and college campuses to the national centers is needed; this will require coordinated investments by all of the stakeholders...." where "... the vast majority of capacity-class computing activities will be carried out on campuses." and "... communication and collaboration across research silos offers the possibility of building and deploying an integrated cyberinfrastructure that effectively and efficiently supports a broad range of scientific and engineering research."

NSF awards to the "Advanced Computing Systems Services" solicitations in 2019 and 2020 guided proposers to take "such considerations as ease of access to proposed systems/services by new communities in S&E; new capabilities that will enable new methods and paradigms for S&E discoveries; and opportunities for leveraging the increasing availability and capabilities at the network edge (including campuses) and via commercial cloud services." [12]

The Community Cluster Program provides a campus-level solution for the usage patterns of today, using batch computing - maximizing utilization, providing economies of scale, giving researchers a solution to not have to "do it themselves", and providing their science problems a necessary avenue of growth from their lab resources, to the campus level, and onward to the national scale.

This pathway from lab to campus to national resources does not yet exist on many campuses for cloud technologies. Faculty use cloud or container technologies like virtualization or Docker in their labs, national solutions like Jetstream [13] or Chameleon [14] exist, and the public cloud provides effectively unlimited scalability at the top end. Many campuses operate administrative computing on shared enterprise virtualization environments, but these systems are not engineered with research use in mind. A new approach is needed to effectively leverage these capabilities to create a coherent campus ecosystem for cloud and container technologies.

Our approach creates new capabilities to empower scientists to integrate cloud computing services into their work, and practice "SciOps" [15] - building on top of DevOps principles like infrastructure as code, automation, version control and continuous integration to make better, more reproducible, and more shareable science.

B. Adapt Community Clustering to a new Paradigm

The Community Cluster Program has successfully demonstrated the value of a partnership between central IT and faculty research groups to acquire, fund, operate, and retire community computing resources to empower faculty in their research and education efforts. We seek to explore the adoption and evolution of this successful model for "Geddes" [16], the Purdue composable community cloud. Our working hypothesis is that is possible to apply the same financial, acquisition, and operations model used for the Community Cluster Program to a composable resource with minimal changes. Our objective is to explore and validate financially and organizationally sustainable models for operating a campus scale community cloud that supports a path from lab scale resources, campus central resources, and potentially external public clouds.

C. Develop the Cyberinfrastructure Workforce

According to "National Strategic Computing Initiative Update: Pioneering the Future of Computing" report [17] (Nov 2019), it is essential to "Create a diverse workforce necessary to achieve the goals of the U.S. Strategic Computing Plan and to support the broader U.S. innovation ecosystem at the leading edge of computation.". Informal anecdotes from the national cyberinfrastructure community about difficulties in recruiting highly skilled candidates for technical positions in support of research projects echo the need described in National Science Technology Council report. There is a need to explore new models to develop a comprehensive training program to increase the skill level and knowledge of the national cyberinfrastructure workforce.

Undergraduate students, especially in STEM disciplines, may not experience an adequate level of hands-on experience with cyberinfrastructure tools and infrastructure to develop a high level of skill in the selection, deployment, and use of cyberinfrastructure and cloud computing technologies. It is essential to increase the skill level of undergraduate students in the application of cyberinfrastructure and cloud computing technologies to solve challenging problems in domain science.

To address this gap, we are applying our structured mentoring program between skilled technical staff and qualified undergraduate students to provide a way to "learn by doing" to boost students' skills. Undergraduate students are key contributors in the deployment and operations of this cyberinfrastructure hardware and software, and benefit from these unique opportunities to apply their classroom knowledge to real-world problems.

Purdue's research computing team has a long track record of utilizing students to support research computing at Purdue [18] [19] [20] [21] [22]. Undergraduate students progress from hardware technicians as underclassmen, to junior system administrators or facilitators by the time they are seniors, and, often, hired into full-time roles. Since 2005, the program has employed over 70 students, with 24% continuing into a research computing career, either at Purdue, another academic institution, or industry.

As described in the recent book "Digital Transformation" [23], there is a need to provide training in new technologies (such as data science) to the national workforce to help society respond to the ongoing digital transformation taking place.

Efforts are underway in Purdue's Polytechnic Institute to investigate the development of an organized education or continuing education program in cyberinfrastructure science for practitioners who already hold a STEM degree. This offering will supplement their existing education and experience to increase the availability of workers skilled in the theory and practice of cyberinfrastructure. Within the research computing community, for practitioners holding a STEM degree, there is a need to supplement their existing education and experience to increase the availability of workers' skills in the theory and practice of cyberinfrastructure. The Geddes composable platform is a tool well-suited for training this digital workforce.

IV. Solution

A. Platform Architecture

The Geddes composable platform uses Rancher [24] to provide a composable infrastructure to launch container-based applications with Kubernetes [25]. Rancher provides a unifying interface to manage multiple Kubernetes clusters with centralized authentication, role-based access control (RBAC), monitoring and alerting in order to create a true multitenant solution.

The system architecture follows a hybrid infrastructure model with hyper-converged servers providing both compute and storage resources as well as a standalone software defined storage solution. Figure 1 shows an architecture diagram.

1) Hyper-converged Application Servers: Application servers have dual AMD Epyc 7662 2.00 GHz 64 core processors (128 cores total), 1 TB of RAM and 24 TB (6x 4TB) of SATA SSDs. Total capacity of eight deployed

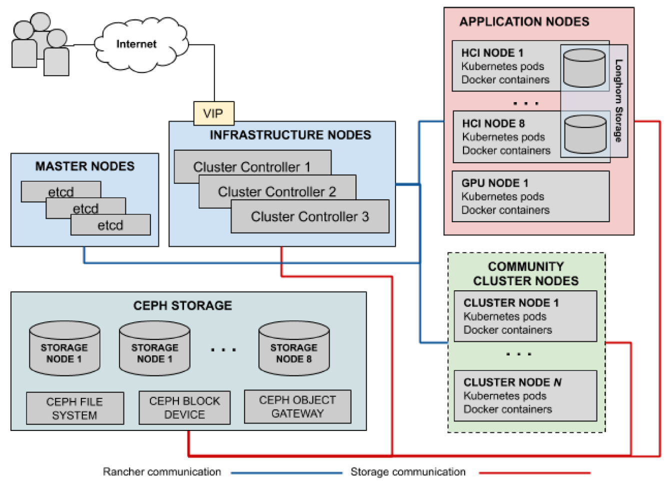

Fig. 1. Geddes Composable Platform Architecture

application servers is 1024 cores and 192 TB of disk space. A planned deployment of GPU servers will incorporate NVIDIA A100 GPUs for machine learning applications. All servers have 100 Gbps network connections.

2) Dedicated Storage Servers: Storage servers have dual Intel Xeon Gold 6126 2.6 GHz 12 core processors (24 cores total), 192 GB RAM and 24 TB (6x 4TB) of NVMe SSDs. Each server connects to SAS disk trays to provide 48TB of capacity storage (12x 4TB hard disks). Eight storage servers provide a total of 192 TB of NVMe storage and 384TB of hard disk storage.

3) Block and Object Storage: To meet the non-POSIX storage demands that come with a composable computing paradigm, we provide multiple methods to access data as objects or block storage. The hybrid infrastructure model allows for flexible and cost effective access to multiple storage tiers: commodity SAS HDD and SATA SSD tiers for general purpose block devices and object stores and a high performance NVMe tier for object storage and block devices for data intensive workloads. SATA based block devices on hyper-converged servers are managed via Longhorn [26]. HDD and NVMe based block and object storage is managed via Rancher with a Rook [27] deployed Ceph [28] instance. The block storage tiers are presented to end users as separate Kubernetes Storage Classes. Using Kubernetes to orchestrate and present the different storage technologies will enable users to get the correct purpose-defined storage to fit their applications.

4) Filesystem Storage: In addition to object or block interfaces, the Ceph storage system provides a shared filesystem for use by private cloud applications. The system also has access to Purdue shared storage resources such as the Research Data Depot [29]. Access to shared storage resources is only allowed via protocols with proper authentication mechanisms (SMB/SSH/Globus). NFS access from Geddes to shared filesystems is not permitted.

5) OS and System Management: Similar to all current Purdue HPC systems, the Geddes platform runs CentOS 7, and uses Extreme Cloud Administration Toolkit (xCAT) [30] and masterless Puppet [31] within Git repositories for provisioning and managing the infrastructure.

6) Monitoring and Alerting: Host level resource monitoring is performed with Sensu Go [32]. All Rancher components, including management and control plane nodes, Kubernetes nodes and workloads are monitored using Prometheus [33]. Alerting is managed in Rancher, using tools built on Prometheus Alertmanager. Alertmanager provides configurable, granular alerting on Prometheus metrics, allowing Purdue staff members to be notified for problems with overall cluster health and allowing cluster users to be notified for problems with their workloads executing in Kubernetes pods. Prometheus metrics and time series data are accessible via Grafana for visualization and reporting.

B. Authentication and Access Control

Users are given access to the Rancher UI via on premise LDAP authentication and managed by Rancher's unique role-based access control policies. Each individual, project, or research group, are assigned a project where they have the ability to manage their own environments. Once users are assigned to a project by an admin, an assigned user owner can be specified. This owner can then manage individual user permissions within their project, alleviating the need for admins to step in for every change needed, unless additional support is requested.

C. Services, Load Balancing and Ingress

Kubernetes provides multiple service types for exposing ports and forwarding traffic to an application: ClusterIP for internal access, NodePort for simple external access via host IPs on a range of ports (30000-32767) and LoadBalancer for directly exposing services via a dedicated IP address. When running Kubernetes on bare metal the LoadBalancer service type is not available by default. We deployed MetalLB [34], a software based load balancing solution, to provide access to LoadBalancer services in Geddes. Users can pick from public and private IP pools depending on their application requirements. An nginx Ingress controller is available for L7 load balancing and virtual hosting.

D. Docker Registries

A centralized managed Docker registry is hosted with custom built and validated images that are commonly used or requested by research computing partners at Purdue. These images, along side Rancher's use of catalogs, provide the end users with the ability to easily and quickly deploy data analysis tools such as Jupyter notebooks and Spark, alongside a wide verity of databases and other data aggregation and enrichment services such as KSQL, PostGIS, Kafka, and ElasticSearch stacks.

A private Docker registry is also available for users who have sensitive data or proprietary code they do not want publicly accessible within their Docker images. This registry is fully integrated and accessible from inside Rancher so that users with UI access can reach and deploy their applications.

E. Security

With any new system, cybersecurity and user privacy is a top priority. Geddes implements a comprehensive yet strict form of typical security methods of firewalls and user access control polices at both the system and application level, while not compromising data confidentiality, integrity, and availability. Additionally, Geddes supports two factor authentication, uses periodic vulnerability scanning from inside and outside of the Kubernetes infrastructure, and monitored by Purdue's PULSAR [35] system, a Zeek based IDS for networking monitoring, logging and alerting. This IDS system monitors all north/south network traffic in and out of the Geddes composable subnets, and is built to expand for monitoring of all east/west traffic as well.

V. Preliminary Results - Science Use Cases

Common use cases for Kubernetes include data ingest nodes running Apache Storm or Kafka, message queue agents for science gateways, and SQL or non-SQL databases for managing datasets—all essential to continually acquiring and processing streaming data from instruments, sensors, and social media and supporting a wide variety of future data analytics applications.

A. RCHE-Hub

The Regenstrief Center for Healthcare Engineering (RCHE) brings together multiple disciplines to collaboratively improve healthcare delivery. This work involves the dynamic sharing of healthcare data to support the refinement and development of proactive and patient-centered healthcare. The tools necessary to support this work include the full breadth of data science capabilities, from Hadoop and NoSQL data stores to browser-based notebooks and analysis applications. Additionally due to the nature of the data, security and availability are key considerations.

The Geddes platform allowed RCHE data scientists to convert from a dedicated, physical infrastructure to establish a dynamic, reproducible replacement where security and compliance considerations were built in from the beginning. The Geddes Rancher control environment is able to control both the controlled cluster hosting the RCHE-hub, and the open research one supporting the rest of campus.

B. Science Gateways

To support FAIR geospatial data, the NSF CSSI-funded GeoEDF gateway [36] provides a cyberinfrastructure to efficiently acquire, manage, and process this sensor data. A private cloud is ideal to host stream processing frameworks like Apache's Kafka where high availability and fault tolerance are important requirements. In order to support efficient retrieval and spatial and temporal filtering, an indexed storage system such as Elasticsearch, mongoDB, or InfluxDB is required. This infrastructure will allow it to be paired with a calibration pipeline that combines current weather data from repositories such as NOAA, and use machine learning to recalibrate imagery obtained at different times and weather conditions. Currently, batch-oriented community cluster resources are not capable of deploying processing frameworks, message queues, or database infrastructure to provide underlying services to science gateways.

Geddes provides the GeoEDF developers a flexible, self-managed infrastructure on which to deploy these supporting services for their science gateway. GeoEDF workflows can utilize both batch HPC and composable steps, in a hybrid model.

C. CoExplorer

One domain science that has seen an exponential increase in demand for cyberinfrastructure is the life sciences. The affordability of next generation sequencing technology has enabled an ever growing number of biologists to generate genome-level data to address their research questions. Applications for analysis and visualization of these data are primarily command-line, but not all scientists wishing to analyze genome-level data have the computer science training to effectively utilize these applications. To overcome this limitation, assistant Professor Jennifer Wisecaver's research team built CoExplorer - a data publication web application that lets users query, visualize, and download gene expression data. CoExplorer is designed to be flexible and applicable to a variety of biological systems. It is written in Python and runs as a Jupyter notebook rendered by Voila [37]. The code and data is packaged into a Docker image for container hosting.

Beyond the specific example of CoExplorer, a Jupyter-Viola-Docker-Kubernetes environment is broadly reusable to allow researchers publish their data with a significantly less developer and hosting overhead than with traditional science gateway platforms. With limited guidance from a full-time developer, investigators can turn their Jupyter notebooks into interactive data publication applications, build images and distribute them to be hosted and shared with their community.

D. High Energy Physics

Particle physics experiments have consistently produced some of the world's largest scientific datasets. Increases in data volume make interactive analysis difficult due to the tremendous time necessary for reading, transforming, and filtering. In the future, new approaches are needed to analyze the ever-growing datasets to continue exploring the nature of the universe. Typical CMS user analysis workflows [38] apply two C++ frameworks to central datasets: CMSSW, for CMS specific analysis, and ROOT, an experiment agnostic toolkit for object serialization and statistical tools. Recently, new tools and systems for analyzing petabyte and exabyte scale datasets based on interactive analysis with Apache Hadoop, Spark and DASK have emerged in industry and proof of concept studies show they also offer promise for analyzing large particle physics datasets. [39]

Geddes is used by the Purdue CMS research team for large-scale data analysis; evolving from traditional batch submission towards interactive services for data analysis built on cloud services [40]; and the use of virtual clusters for rapid development.

VI. Future Work

This composable system is built in such a way that it can be dynamically expanded by reallocating community cluster HPC compute nodes into Kubernetes nodes using the research computing team's xCAT management tools. Future effort will make this capability seamless rather than manual, and will allow the composable platform to become truly dynamic.

A. Business Model

As the first researchers begin to achieve scientific results from their use of the Geddes composable platform, a key question to be answered - and best practices shared with the national community - is a reusable business model for such a resource.

There are several potential specific strategies under consideration for a cost structure.

- Public cloud-like hourly charges for containers, GB, etc.
- Annual subscription prices for effectively unlimited access to the system
- A model like the community cluster model where a researcher may purchase a unit of capacity and run as many containers, or GB of storage as they can fit
- Or a model that allows researchers to dynamically shift their investment between (HPC) community clusters, and the composable platform.
- Or, some hybrid of the above.

Feedback from Purdue researchers, cost accounting, and experiences from other cloud operators such as RedCloud [41], Jetstream [13], and Minnesota [42] will be critical in making a successful business model.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 2018926. 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 National Science Foundation.

References

[1] T. Sterling, D. J. Becker, D. Savarese, J. E. Dorband, U. A. Ranawake, and C. V. Packer, "Beowulf: A Parallel Workstation For Scientific Computation," in *In Proceedings of the 24th International Conference on Parallel Processing*. CRC Press, 1995, pp. 11–14.

[2] G. McCartney, T. Hacker, and B. Yang, "Empowering Faculty: A Campus Cyberinfrastructure Strategy for Research Communities," *Educause Review*, 2014.

[3] "ITaP Research Computing - Data Workbench." [Online]. Available: https://www.rcac.purdue.edu/compute/workbench

[4] "What is composable infrastructure? — Network World." [Online]. Available: https://www.networkworld.com/article/3266106/what-is-composable-infrastructure.html

[5] A. G. Carlyle, S. L. Harrell, and P. M. Smith, "Cost-Effective HPC: The Community or the Cloud," *IEEE Conference on Cloud Computing Technology and Science*, dec 2010.

[6] P. Smith, S. L. Harrell, A. Younts, and X. Zhu, "Community Clusters or the Cloud: Continuing Cost Assessment of On-premises and Cloud HPC in Higher Education," in *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (Learning)*, ser. PEARC '19. New York, NY, USA: ACM, 2019, pp. 97:1—-97:4. [Online]. Available: http://doi.acm.org/10.1145/3332186.3333155

[7] A. Chalker, C. W. Hillegas, A. Sill, S. Broude Geva, and C. A. Stewart, "Cloud and on-premises data center usage, expenditures, and approaches to return on investment: A survey of academic research computing organizations," in *Practice and Experience in Advanced Research Computing*, ser. PEARC '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 26–33. [Online]. Available: https://doi.org/10.1145/3311790.3396642

[8] S. Rajamohan and R. E. Settlage, "Informing the on/off-prem cloud discussion in higher education," in *Practice and Experience in Advanced Research Computing*, ser. PEARC '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 64–71. [Online]. Available: https://doi.org/10.1145/3311790.3396627

[9] D. Thain, T. Tannenbaum, and M. Livny, "Distributed computing in practice: the Condor experience." *Concurrency - Practice and Experience*, vol. 17, no. 2-4, pp. 323–356, 2005.

[10] D. Merkel, "Docker: Lightweight Linux Containers for Consistent Development and Deployment," *Linux J.*, vol. 2014, no. 239, mar 2014. [Online]. Available: http://dl.acm.org/citation.cfm?id=2600239.2600241

[11] D. A. Bader, G. Tech, H. M. Berman, M. Parashar, T. H. Dunning, G. Jacobs, T. Agerwala, M. Hildreth, R. Loft, and S. Ruggles, "CI2030: Future Advanced Cyberinfrastructure A report of the NSF Advisory Committee for Cyberinfrastructure Transmittal letter from the NSF Office of Advanced Cyberinfrastructure Report CI2030: Future Advanced Cyberinfrastructure NSF Advisory Committee for Cyberinfrastructure 1 Cyberinfrastructure Ecosystem Working Group," Tech. Rep., 2018. [Online]. Available: https://www.nsf.gov/cise/oac/ci2030/.

[12] "Advanced Computing Systems & Services: Adapting to the Rapid Evolution of Science and Engineering Research (nsf19587) — NSF – National Science Foundation." [Online]. Available: https://www.nsf.gov/pubs/2019/nsf19587/nsf19587.htm

[13] C. A. Stewart, T. M. Cockerill, I. Foster, D. Hancock, N. Merchant, E. Skidmore, D. Stanzione, J. Taylor, S. Tuecke, G. Turner, M. Vaughn, and N. I. Gaffney, "Jetstream: A self-provisioned, scalable science and engineering cloud environment," in *Proceedings of the 2015 XSEDE Conference: Scientific Advancements Enabled by Enhanced Cyberinfrastructure*, ser. XSEDE '15. New York, NY, USA: Association for Computing Machinery, 2015. [Online]. Available: https://doi.org/10.1145/2792745.2792774

[14] 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)*. USENIX Association, July 2020.

[15] I. Jimenez, "AGILE RESEARCH DELIVERY," Ph.D. dissertation, UC Santa Cruz, 2019.

[16] "Geddes, LaNelle E. - Purdue University Archives." [Online]. Available: https://archives.lib.purdue.edu/agents/people/802

[17] F.-t. O. Action Committee Strategic Computing, Tech. Rep.

[18] M. E. Baldwin, X. Zhu, P. M. Smith, S. L. Harrell, R. Skeel, and A. Maji, "Scholar: A campus hpc resource to enable computational literacy," in *2016 Workshop on Education for High-Performance Computing (EduHPC)*, Nov 2016, pp. 25–31.

[19] S. Harrell and A. Howard, "Hybrid HPC Cloud Strategies from the Student Cluster Competition," in *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*. IEEE, jul 2018, pp. 186–193. [Online]. Available: https://ieeexplore.ieee.org/document/8457799/

[20] E. Hillery, M. D. Ward, J. Rickus, A. Younts, P. Smith, and E. Adams, "Undergraduate Data Science and Diversity at Purdue University," in *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (Learning)*, ser. PEARC '19. New York, NY, USA: ACM, 2019, pp. 88:1—-88:6. [Online]. Available: http://doi.acm.org/10.1145/3332186.3332202

[21] S. L. Harrell, P. M. Smith, D. Smith, T. Hoefler, A. A. Labutina, and T. Overmyer, "Methods of creating student cluster competition teams," in *Proceedings of the 2011 TeraGrid Conference: Extreme Digital Discovery*. ACM, 2011, p. 50.

[22] A. Howard, A. Younts, P. M. Smith, and J. J. Evans, "Undergraduate experience in clustering at the SC07 Cluster Challenge," in *In Proceedings of the 2008 Linux Clusters Institue*, 2008.

[23] T. Siebel, *Digital Transformation: Survive and Thrive in an Era of Mass Extinction*. New York, NY: Rosetta Books, 2019.

[24] "Enterprise Kubernetes Management — Rancher." [Online]. Available: https://rancher.com/

[25] B. Burns, B. Grant, D. Oppenheimer, E. Brewer, and J. Wilkes, "Borg, Omega, and Kubernetes," *Queue*, vol. 14, no. 1, pp. 70–93, jan 2016. [Online]. Available: https://doi.org/10.1145/2898442.2898444

[26] "Longhorn." [Online]. Available: https://longhorn.io/

[27] "Rook." [Online]. Available: https://rook.io/

[28] S. A. Weil, S. A. Brandt, E. L. Miller, D. D. E. Long, and C. Maltzahn, "Ceph: A Scalable, High-Performance Distributed File System," in *Proceedings of the 7th Symposium on Operating Systems Design and Implementation*, ser. OSDI '06. USA: USENIX Association, 2006, pp. 307–320.

[29] D. T. Dietz, L. A. Gorenstein, G. S. Veldman, and K. D. Colby, "Shared Research Group Storage Solution with Integrated Access Management," in *Proceedings of the Practice and Experience in Advanced Research Computing 2017 on Sustainability, Success and Impact*, ser. PEARC17. New York, NY, USA: ACM, 2017, pp. 14:1–14:7. [Online]. Available: http://doi.acm.org/10.1145/3093338.3093354

[30] E. Ford, B. Elkin, S. Denham, B. Khoo, M. Bohnsack, C. Turcksin, and L. Ferreira, "Building a Linux HPC Cluster with xCAT," *IBM Redbook*, 2002.

[31] J. St. John, "xCAT and Masterless Puppet: Aiming For Ideal Configuration Management," in *Proceedings of the HPC Systems Professionals Workshop*, Dallas, TX.

[32] "Sensu." [Online]. Available: https://sensu.io/

[33] "Prometheus - Monitoring system time series database." [Online]. Available: https://prometheus.io/

[34] "MetalLB, bare-metal load balancer for Kubernetes." [Online]. Available: https://metallb.universe.tf

[35] S. Trivedi, L. Featherstun, N. DeMien, C. Gunlach, S. Narayan, J. Sharp, B. Werts, L. Wu, C. Ellis, L. Gorenstein, E. Gough, A. Younts, and X. Zhu, "Pulsar: Deploying network monitoring and intrusion detection for the science dmz," in *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (Learning)*, ser. PEARC '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: https://doi.org/10.1145/3332186.3333058

[36] R. Kalyanam, L. Zhao, X. C. Song, V. Merwade, J. Jin, U. Baldos, and J. Smith, "Geoedf: An extensible geospatial data framework for fair science," in *Practice and Experience in Advanced Research Computing*, ser. PEARC '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 207–214. [Online]. Available: https://doi.org/10.1145/3311790.3396631

[37] "Voila Dashboards." [Online]. Available: https://github.com/voila-dashboards/voila

[38] D. Bonacorsi, C. M. S. Collaboration, and Others, "The CMS computing model," *Nuclear Physics B-Proceedings Supplements*, vol. 172, pp. 53–56, 2007.

[39] O. Gutsche, L. Canali, I. Cremer, M. Cremonesi, P. Elmer, I. Fisk, M. Girone, B. Jayatilaka, J. Kowalkowski, V. Khristenko, E. Motesnitsalis, J. Pivarski, S. Sehrish, K. Surdy, and A. Svyatkovskiy, "CMS Analysis and Data Reduction with Apache Spark," 2017.

[40] L. Bauerdick, B. Bockelman, D. Dykstra, S. Fuess, G. Garzoglio, M. Girone, O. Gutsche, B. Holzman, D. Hufnagel, H. Kim, R. Kennedy, D. Mason, P. Spentzouris, S. Timm, A. Tiradani, and E. Vaandering, "Experience in using commercial clouds in CMS," *Journal of Physics: Conference Series*, vol. 898, p. 052019, oct 2017. [Online]. Available: http://stacks.iop.org/1742-6596/898/i=5/a=052019?key=crossref.fdacb443871e9d5ce74ef50a7d4ee629

[41] R. Knepper, S. Mehringer, A. Brazier, B. Barker, and R. Reynolds, "Red cloud and aristotle: Campus clouds and federations," in *Proceedings of the Humans in the Loop: Enabling and Facilitating Research on Cloud Computing*, ser. HARC '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: https://doi.org/10.1145/3355738.3355755

[42] E. F. Bollig and J. C. Wilgenbusch, "From bare metal to virtual: Lessons learned when a supercomputing institute deploys its first cloud," in *Proceedings of the Practice and Experience on Advanced Research Computing*, ser. PEARC '18. New York, NY, USA: Association for Computing Machinery, 2018. [Online]. Available: https://doi.org/10.1145/3219104.3219164