Anvil - System Architecture and Experiences from Deployment and Early User Operations

Carol X. Song
Purdue University
West Lafayette, USA

Preston Smith
Purdue University
West Lafayette, USA

Rajesh Kalyanam
Purdue University
West Lafayette, USA

Xiao Zhu
Purdue University
West Lafayette, USA

Eric W. Adams
Purdue University
West Lafayette, USA

Kevin D. Colby
Purdue University
West Lafayette, USA

Patrick T. Finnegan
Purdue University
West Lafayette, USA

Erik Gough
Purdue University
West Lafayette, USA

Elizabeth Hillery
Purdue University
West Lafayette, USA

Rick Irvine
Purdue University
West Lafayette, USA

Amiya K. Maji
Purdue University
West Lafayette, USA

Jason St. John
Purdue University
West Lafayette, USA

ABSTRACT

Anvil is a new XSEDE advanced capacity computational resource funded by NSF. Designed with a systematic strategy to meet the ever increasing and diversifying research needs for advanced computational capacity, Anvil integrates a large capacity high-performance computing (HPC) system with a comprehensive ecosystem of software, access interfaces, programming environments, and composable services in a seamless environment to support a broad range of current and future science and engineering applications of the nation's research community. Anchored by a 1000-node CPU cluster featuring the latest AMD EPYC 3rd generation (Milan) processors, along with a set of 1TB large memory and NVIDIA A100 GPU nodes, Anvil integrates a multi-tier storage system, a Kubernetes composable subsystem, and a pathway to Azure commercial cloud to support a variety of workflows and storage needs. Anvil was successfully deployed and integrated with XSEDE during the world-wide COVID-19 pandemic. Entering production operation in February 2022, Anvil will serve the nation's science and engineering research community for five years. This paper describes the Anvil system and services, including its various components and subsystems, user facing features, and shares the Anvil team's experience through its early user access program from November 2021 through January 2022.

CCS CONCEPTS

• **Computer systems organization**; • **Hardware**; • **Computing methodologies** → *Parallel computing methodologies*; *Distributed computing methodologies*;

1 INTRODUCTION

The demand for diverse modalities of computational capacity is increasing more rapidly than the current national advanced computing infrastructure - XSEDE [15, 16] - can accommodate primarily due to the convergence of two trends: traditionally computing intensive domains are growing exponentially, and new and more diverse research domains are emerging as major computational resource users. This convergence highlights three main challenges to meeting the rapidly evolving landscape of computational needs: (1) the ever-increasing need for computational resources in almost all domains of science and engineering (S&E); (2) the new computational paradigms (e.g., non-batch computing, AI/ML) and expansion to non-traditional HPC domains, and (3) training the next generation researchers and workforce for cyberinfrastructure (CI) sustainability.

Employing a systematic strategy to meeting these needs, a Purdue University Rosen Center for Advanced Computing team has developed and deployed Anvil, a new national advanced computational resource, and a broad range of capabilities and services. Funded by the NSF, Anvil combines a large-capacity high-performance computing cluster with a comprehensive ecosystem of software, access interfaces, programming environments, and composable services to form a seamless environment able to support a broad range of current and future S&E applications. Anvil is integrated with XSEDE and has been operating and supporting users since November 1, 2021. To date nearly 117M CPU service units (SUs) and 290K GPU SUs have been allocated on Anvil through the

Component	Features
Standard Compute	1000 128-core Dell compute nodes, with 3rd Generation AMD Epyc processors, with a peak performance of 5.1 PF
Large Memory Compute	32 large memory compute nodes each with 1 TB of memory
GPU Compute	A total of 64 Nvidia A100 GPUs providing an additional 1.5 PF of single-precision performance
Interconnect	HDR100 Infiniband, 3:1 fat tree
Data Storage	Arcastream Pixstor - 10 PB DDN storage, (disk tier), 3 PB of Dell storage (flash tier)
Composable System	8 composable compute nodes, managed by Kubernetes

Table 1: Anvil System Specifications

quarterly XSEDE allocation process (XRAC). Anvil also provides nearly 20M CPU and 30K GPU SUs to researchers as part of the COVID-19 HPC Consortium. In the following sections we describe how the increasing need for computational resources has driven the Anvil system architecture, the various innovative capabilities that support new computational paradigms and non-traditional HPC users and domains, as well as our ongoing and planned workforce development activities to train the next generation CI workforce.

2 ARCHITECTURE

The Anvil system design was motivated by two main considerations. Firstly, Anvil is to deliver a large quantity of computing power to support a large number of current and future science and engineering applications. Based on our analysis of workloads on XSEDE and Purdue clusters, Anvil is optimized to support moderate-sized workloads (e.g., under 1024 cores) while also allowing wider jobs to run on the system. Anvil's 128-core compute nodes were chosen after analysis of XSEDE usage data, which revealed that such a node could support significant fractions of today's workloads on XSEDE. The expected major growth in Python, R, notebooks and AI applications has led to the optimization of Anvil for both traditional and emerging scientific workloads. Secondly, recognizing the changing application landscape on HPC systems where complex workflows, data-driven computations, and interactive explorations at all scales are becoming more common, Anvil complements its core HPC system with composable edge services, a variety of storage types and access interfaces to meet these needs and make Anvil more accessible to a broader audience. These considerations have impacted Anvil's design decisions, including the selection of the type of nodes, interconnect, queues, storage types, and the software ecosystem.

Built in partnership with Dell, Anvil consists of the following components, summarized in Table 1.

Compute: The Anvil compute system consists of 1000 base compute nodes based on the AMD 3rd generation EPYC 7763 64-core (Milan) processor. Each compute node features two 7763 CPUs, and 256 GB of RAM. The CPU partition delivers up to 5.1 TeraFLOPS (TF) of peak performance per node, for a total of 5.1 PetaFLOPS (PF).

The base compute nodes are complemented by 32 large-memory nodes with 1 TB of RAM each, and 16 GPU compute nodes, each

with 4 NVIDIA A100 GPUs. The GPU subsystem provides a total of 64 GPUs, with 1.5 PF of single-precision performance.

Storage: The primary file storage platform is a 13 PB raw (11 PB usable) parallel filesystem based on Arcastream's Pixstor product, utilizing disk storage from DDN and flash storage from Dell. Based on Spectrum Scale, Pixstor provides a multi-tier filesystem utilizing flash to accelerate writes and actively read files, and provides a policy engine to migrate data to the large-capacity disk tier.

The Pixstor filesystem also provides persistent file storage for active allocations, and a repository for commonly used datasets. Anvil users can leverage Purdue's HPSS-based Fortress tape archive for longer-term storage for active allocations. Anvil's home directories and application software are provided by a dedicated ZFS file server with a total capacity of 51 TB.

Composable Subsystem: Anvil's composable subsystem is based on 8 large memory compute nodes with 1 TB of RAM, and 6 TB of local storage. The composable subsystem is managed via Kubernetes [3], with Rancher [14] software providing a GUI-based control plane and Ceph [18] providing block, filesystem and object storage.

3 INNOVATIVE CAPABILITIES

Anvil offers a number of innovative capabilities in response to the changing landscape of research applications and HPC user base. Traditional HPC users benefit from powerful tools like container distributions for easy access to applications, while nontraditional HPC users and training activities can take advantage of interactive computing capabilities that reduce the barrier to entry to HPC, and leverage composable technologies that complement Anvil's batch computing capabilities.

3.1 Software Ecosystem

Anvil provides a diverse suite of software including compilers, debuggers, visualization libraries, development environments, and other commonly used application software. The Spack [7] package management tool is used to manage multiple library versions and configurations for ease of compilation and continuous deployment. Lmod and the module system are used to manage the user environment, allowing users to easily load the requisite software and dependencies while also ensuring the appropriate Linux environment variables are set. Anvil also provides "canonical" CPU
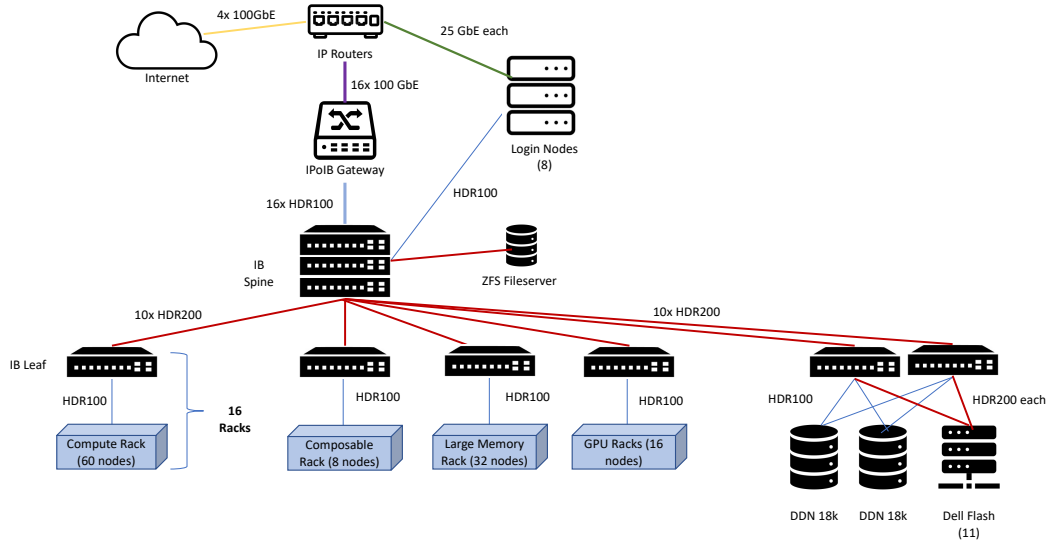
Figure 1: Anvil System Architecture

and GPU software stacks in the form of a module that is loaded by default on Anvil nodes. The CPU module comprises a default compiler and MPI environment, while the GPU module comprises a stable CUDA version, compiler, and math library.

The Anvil software ecosystem also consists of tools that are geared towards modern machine learning, data science, and bioinformatics applications. Containerization has been leveraged by projects such as NVIDIA GPU Cloud (NGC) [13] and BioContainers [6] to distribute heavily optimized and efficient software tools for these applications. Anvil hosts a subset of these containers, including more than 400 biocontainers, that can be loaded via the module system, while other containers available on these project sites can be downloaded and run via Anvil's Singularity container platform.

3.2 Interactive Computing

Using the Cendio ThinLinc [4] software pioneered in Purdue's Data Workbench [5] system, users can quickly log into Anvil and launch common HPC science applications as batch jobs from the convenience of a desktop menu. Menu applications also include easy-to-use graphical tools to launch Windows VMs to use non-Linux software. ThinLinc is well-suited to remote visualization applications even over slow or high-latency networks.

Anvil uses the Open OnDemand [12] software developed by the Ohio Supercomputer Center to provide another mode of easy access to science applications. Open OnDemand provides browser-based file management, job templates, and gateway-style application hosting. The applications published in OnDemand include R Studio, Jupyter notebooks, Spark, MATLAB, and more.

3.3 Composable Subsystem

Anvil features a composable subsystem [9] to dynamically provision infrastructure from pools of computing resources (CPU, memory,

disk, network). Rancher is used to provide a control plane to the composable infrastructure to launch container-based applications via Kubernetes [3]. Using an intuitive user interface or command line tools, users can deploy containerized applications, science gateways, elastic software stacks, and data analysis pipelines to complement batch HPC workflows. Common use cases for the composable subsystem include SQL or NoSQL databases, personal science gateways and parallel, scalable analysis frameworks based on Dask or Spark.

The Microsoft Azure cloud will be integrated to facilitate bursting to the cloud for composable workflows or appropriate HPC applications. During the deployment phase, Azure was used for early access to compute nodes for benchmarking and staff training[20]. Anvil HPC bursting [17] will utilize HBv3 instances built on equivalent 128-core AMD processors.

4 WORKFORCE DEVELOPMENT, TRAINING & COMMUNITY ENGAGEMENT

The Anvil project has developed a comprehensive plan for CI workforce development. It builds upon and expands Purdue's existing programs in mentoring future CI professionals, training for computational and data literacy, and supporting under-served communities. Since 2003, undergraduates from disciplines such as computer science, information technology and multiple engineering fields have been mentored by Purdue's research computing staff through internships and student programs in hardware maintenance, systems administration, HPC tool development, and science gateway development [2] [10] [11]. The Purdue student programs have mentored more than 80 students since 2008, with over 25% of them moving on to a career in academic research computing or HPC industry.

The Anvil team is launching a new REU program in the summer of 2022 to extend the Purdue CI professional apprenticeship program to students nationwide, with a particular emphasis on under-served communities such as women [1], minorities, and under-resourced institutions. The 12-week Anvil REU program recruits nationwide and pairs each student with a CI professional mentor to work on a range of R&D projects that will directly impact the Anvil project and the broader CI community. In the first year, the student projects focus on improving system data collection and reporting, application monitoring, container and application cybersecurity, and impact assessment of Anvil's novel interactive and cloud computing tools. Future projects are designed to train students in researcher-facing roles, gateway support, and deeper explorations of accelerator, cloud and composable technologies. Students who complete this internship will be prepared to enter the Research Computing and Data (RCD) workforce to support CI resources and their users.

The team also engaged with the research computing communities through presentations at CASC, PEARC, and Supercomputing conferences, and webinars to various user and CI facilitator groups such as the Campus Champions and CaRCC. An early tutorial at PEARC21 on the composable system was conducted to prepare the science and engineering community for the novel capabilities that Anvil brings to the XSEDE ecosystem [8]. A training curriculum with progressing tiers has been created to meet the diverse user needs. For instance, the introductory level courses (Anvil 101) include basics of Linux, running jobs, using storage, using interactive access interfaces, and HPC seminar series for undergraduate students. The advanced tutorials include Anvil 200 short courses on Kubernetes, object storage with Ceph, archiving data, and cloud bursting to Azure. More targeted training will cover specific tools such as Jupyter Kernels & HPC and Using GPUs with Python. This curriculum will continue to evolve in response to community input and will also leverage learning materials from other projects such as those funded by the OAC CyberTraining program.

As a new system in the national CI ecosystem, the Anvil team reached out to the broader communities to build its user base and enable knowledge sharing through multiple venues. During the acquisition and early user phases, the team engaged in technology forums and partnerships with industry such as AMD and Microsoft, federal agencies via NITRD (Networking and Information Technology Research and Development Program), and various academic research computing centers, sharing our experiences as early adopters of the newer hardware and systems and getting community feedback on best practices. Several staff members from Purdue played leadership roles in the AMD HPC user forum community and shared insights on diverse topics including AMD hardware benchmarking results, Spack automation framework stability, application build related issues and performance issues with peer institutions. These interactions helped shape the choice of compiler versions and application configurations on Anvil. The Anvil leadership team presented to NITRD's member agencies on today's HPC vendor ecosystem from the perspective of a campus. These interactions helped the team learn about the national exascale computing landscape and share the campus perspective on the critical problems facing the researchers of tomorrow.

5 SYSTEM DEPLOYMENT AND PERFORMANCE

The Anvil team has successfully acquired, assembled and deployed the system and conducted an early user program in 2021 all through a world-wide pandemic. The Anvil acquisition has been accepted by NSF and formally entered production operations in February 2022.

From January through July of 2021, the Anvil system was ordered, received and deployed in the Purdue research data center. During the winter of 2020-21, Purdue engineers relocated or decommissioned administrative computing systems to make room for Anvil, deployed Anvil's new liquid cooling infrastructure, and began deploying compute nodes during June of 2021. The entire system was fully running by early fall 2021.

During the fall of 2021, Anvil was put through a number of acceptance benchmarks. These tests included single-node HPL, measurement of the InfiniBand performance with the OSU benchmark suite, STREAM memory tests, and application and system-wide benchmarks. The following sections describe the benchmarking results and lessons learned in more details.

5.1 Application Benchmarks

Anvil was designed to support a wide variety of S&E applications seen at Purdue (as a representative campus) and XSEDE today. The top applications being run on Purdue's campus HPC systems include molecular dynamics (LAMMPS, GROMACS), material modeling (VASP), computational fluid dynamics (OpenFOAM, GEMS), high-energy physics, and weather modeling (WRF, CESM). XSEDE data for 2017-19 also reveals similar categories and software in the top 30 applications: molecular dynamics (NAMD, LAMMPS, GROMACS, Amber), materials modeling (Quantum Espresso), computational fluid dynamics (OpenFOAM, NEK5000), and weather modeling (WRF, CESM).

The Anvil team expected to support a similar mix of applications for traditional HPC users today, as well as new and long-tail applications for the future, with major growth anticipated in Python, R, notebooks, and AI applications. These applications were also run during benchmarking and the early operations phase to demonstrate real-world utilization patterns.

5.1.1 Single Node Application Benchmarks. Figure 2a describes the single node application and benchmark performance obtained on both Anvil and Azure HBv3 instances, shown with benchmark results performed on the 2nd generation AMD EPYC 7742 processor (Rome) as a baseline. All Anvil application benchmarks exceeded or were close to projected performance, and exceeded the single-node performance observed on Azure HBv3.

5.1.2 Memory Bandwidth. Table 2 summarizes the results of STREAM benchmarks performed on Anvil during acceptance testing. Memory bandwidth on Anvil exceeded the projections based on earlier platforms. As expected, this has shown to result in a significant performance improvement on scientific applications running on Anvil during the early user phase and in production today.
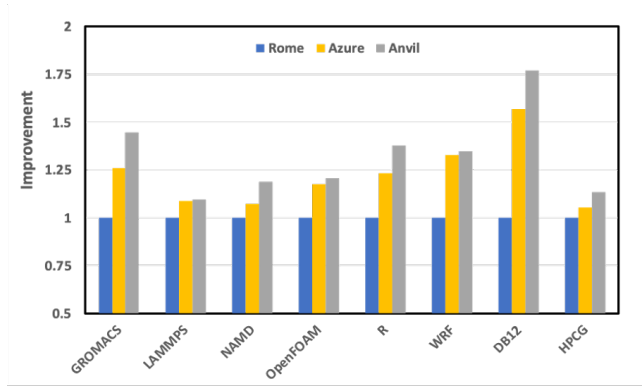
(a) Anvil Single Node Benchmarks
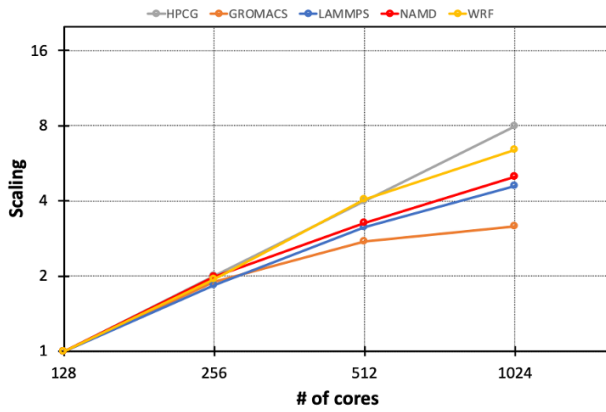
(b) Anvil Strong Scaling Study

Figure 2: Anvil benchmark and scaling studies

STREAM Benchmark	Performance on Sky Lake	Performance on Rome	Projection on Milan	Actual on Anvil
COPY	144064.5	203522.5	284931.5	345981.3
SCALE	143006.6	196341.9	247878.7	346168.7
ADD	148988.5	215621.8	301870.5	347385.5
TRIAD	148831.5	223098.8	312338.3	347375.4

Table 2: Anvil STREAM results

5.1.3 Strong Scaling Study. Strong scaling study of these representative applications also shows good scalability on Anvil up to modest scale (1024 cores/ 8 nodes). (Figure 2b)

5.1.4 GPU Performance. Performance of each GPU (NVIDIA A100 SXM4, 40GB) on Anvil was evaluated by measuring FP32 throughput (number of training images processed per second) while training models on synthetic data. The benchmark shows a 2X speedup on Anvil compared to NVIDIA V100 GPUs for the training of ResNet-50 V1.5 (Figure 3).

5.1.5 Storage Performance. Aggregate performance of the Anvil storage system was measured with mdtest and IOR. To simulate system-wide load, multiple IOR benchmarks were run simultaneously against the scratch tier (flash) and project tier (disk). These
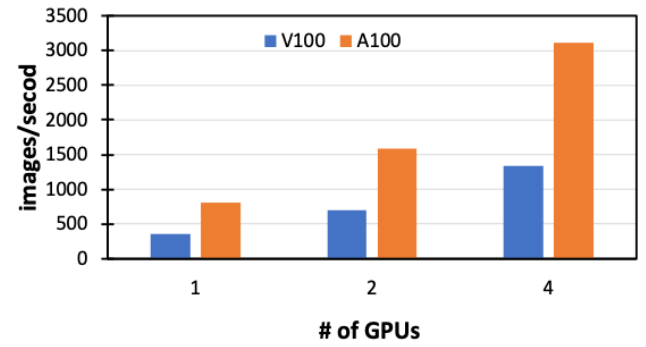

Figure 3: Anvil GPU ResNet benchmark

benchmarks demonstrate an aggregate performance of 165.47 GB/s reads, and write performance of 135.05 GB/s.

5.2 Lessons Learned

Extending the Anvil team's expertise from building campus CI to developing a national CI resource was a valuable experience and here we summarize several key lessons learned from Anvil's deployment.

(1) **XSEDE Integration** Many of the challenges encountered were in how to link Anvil with central XSEDE management services. Starting this integration very early during the deployment phase allowed time to address these issues.

(a) *Domain blending:* XSEDE Anvil user accounts originate with and are managed by XSEDE and thereby do not go through the standard campus workflow for account creation and management. Nevertheless, we wished to continue to use centralized campus accounts for administrators and a small portion of Purdue projects on Anvil which would be outside the scope of XSEDE. A new approach to LDAP tree management and creation for Anvil had to devised in order to allow for Anvil to host a blended set of users from both XSEDE and campus authorities. Care had to be taken to negotiate naming schemes which would avoid eventual collisions in the username, group name, UID, and GID namespaces between XSEDE and campus.

(b) *AMIE integration:* XSEDE resource provider systems must run an AMIE client in order to exchange packets with XSEDE and properly create and link local resource user accounts with the central XSEDE user accounts. The decades-old AMIE software and protocol has been extended to work for XSEDE in a number of ways over the years but has become convoluted and poorly understood. The current AMIE version for XSEDE also lacks any working client implementation. It took several months of work to get a working AMIE client implemented for Anvil, and much of this was due to misunderstandings about how AMIE worked and not encountering many of the protocol scenarios until well after we started receiving production packets from XSEDE.

(c) *Usage reporting:* Usage reporting for XSEDE has moved to a new, custom XSEDE usage reporting protocol which

only one other resource provider is known to have used. Developing this client for Anvil required addressing Slurm job log idiosyncrasies when translating Slurm records for submission to the XSEDE service. We also encountered some transparency and scaling issues with the XSEDE service itself. XSEDE development staff for the new usage reporting system have been quite responsive to this though, and have already started to implement new APIs to allow insight into what the reporting system is doing with uploaded data, and restructuring the service to avoid these scaling failures. These new APIs developed by XSEDE as part of the Anvil integration will make future usage reporting from other resource providers much easier, and the Anvil client developed will be shared across the XSEDE community as well.

(2) **Scale Considerations** One of the key lessons learned during the deployment of Anvil is how to detect and address scalability issues. Purdue's campus community clusters are typically composed of 500-600 compute nodes, whereas, Anvil consists of 1048 compute nodes. This 2x increase in cluster size led to a variety of subtle issues which never manifested in campus clusters. We highlight three of these issues and our design choices here:

(a) *Reducing system image size:* The compute nodes on Anvil are provisioned using the xCAT provisioning software which boots nodes by downloading appropriate OS images over the management network. However, since the management network is much slower compared to the 100Gbps InfiniBand fabric, this has the potential for network congestion when booting the entire cluster from a cold state. To avoid this, we carefully removed all unnecessary RPMs (such as multimedia and graphics libraries) from the system images without reducing Anvil's usability or its interactive compute capability. Compared to Purdue's campus clusters, the Anvil compute images are reduced by 50% or more in size.

(b) *Choice of subnet manager (SM):* During early deployment of Anvil, we used the embedded subnet manager on one of the switches on the Anvil fabric. However, as we racked and powered on the entire cluster, a subset of the nodes could not join the fabric. After investigation, we found that the embedded subnet manager had a maximum capacity of 1024 InfiniBand ports which is lower than Anvil's cluster size. We therefore migrated the subnet manager to a software-based SM, or `opensm` which allowed us to connect all the nodes to the fabric without further issues.

(c) *Defining ARP table size:* The Linux kernel uses ARP (Address Resolution Protocol) tables to map IP addresses to corresponding MAC addresses. However, the default size of ARP tables in Linux is set to 1024. This initially resulted in instability of the GPFS filesystem for Anvil as the storage servers could only connect to a subset of the compute nodes. To address this issue, we increased the maximum size of the ARP cache 8x by setting appropriate kernel parameters: `net.ipv4.neigh.default.gc_thresh3 = 8192`.

6 EARLY USER OPERATIONS

The Anvil Early User Program (EUP) was established in the spring of 2021 and applications were solicited through various venues. In total 39 applications were received out of which 29 were granted. The early users include a diverse group, representing researchers from 22 institutions, 17 fields of science, including 31% who have not had XSEDE allocations previously. The first cohort of early users started using Anvil on 10/31/21 and the formal EUP was concluded on 12/31/21. Nearly 60% of the user groups requested CPU resources alone, while the rest requested a mix of CPU and GPU resources.

6.1 System Usage during EUP

During the two month EUP, 24 early user groups submitted a total of 144, 662 jobs on Anvil and consumed a total of 9.8M CPU service units and 42.7K GPU service units. The observed job sizes bear out our CPU and GPU node design choices. Specifically, a majority of CPU jobs requested fewer than 128 cores and more than 99% of the jobs used fewer than 1024 cores. The top five largest individual jobs (in terms of CPU cores) used between 2100 and 7200 cores which still fit within one rack on Anvil. A majority of GPU jobs requested either 1 or 4 GPUs. System utilization saw a steady increase during the two month EUP period, peaking at nearly 45% in the second half of December 2021.

6.2 Science Accomplishments

The computations carried out on Anvil during the early user period also enabled tangible science accomplishments in several domains. We report on the accomplishments of four such early user groups.

6.2.1 Astronomy and Planetary Sciences. This group seeks to develop numerical tools to model obliquely rotating magnetic massive star winds, and a rare (and only known) magnetic massive star binary system, ϵ Lupi. The developed models utilize full 3D magnetohydrodynamics(MHD) methods that require over 64,000 core hours for a single successful model run. The group was able to carry out a model run for nearly 83,000 core hours and visualize the results of model evolution at various time snapshots. The visualizations (Figure 4) helped the researcher confirm episodic centrifugal mass ejection and the expected slow velocity of dense wind material flow.

6.2.2 Bioinformatics. This early user sought to carry out analysis of 648 human whole-genome bisulfite sequencing (WGBS) samples created as a part of a research protocol involving DNA methylation patterns in psychiatric syndromes such as PTSD. This involved running a toolchain of several well-known bioinfomatics tools, including a GPU-accelerated short-read alignment tool created by this researcher. Computations on Anvil enabled the researcher to evaluate the performance of their GPU-accelerated tool on the A100 GPUs, and achieve much better throughput (Figure 5) when compared to other XSEDE systems due to a combination of the A100 performance and improved filesystem (GPFS) I/O speeds.

6.2.3 Computer-aided Drug Design. GPCRs serve as the primary targets of $\sim 1/3$ of the currently marketed drugs; however the molecular mechanisms by which GPCRs catalyze the G protein
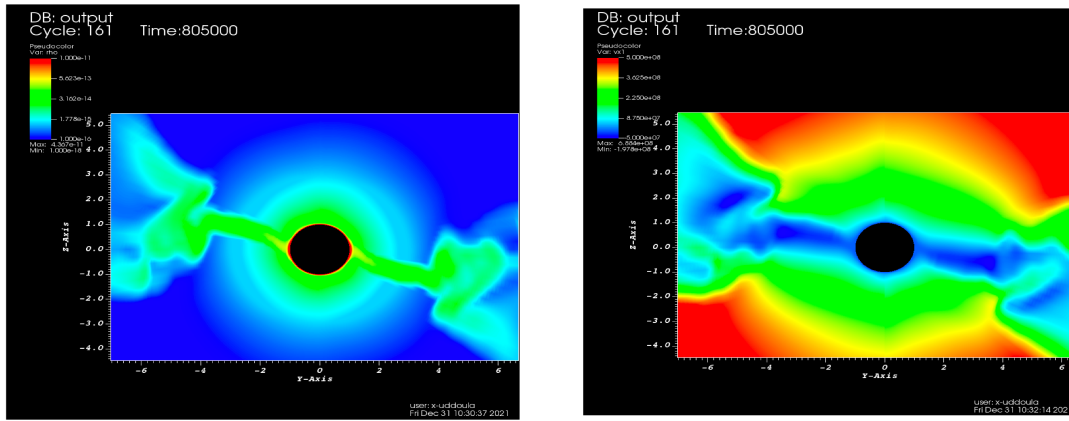
Figure 4: Slices through Z-plane from the 3D MHD simulation, logarithm of density (left), radial velocity (right) at the 800 ks physical time snapshot. (Images courtesy of Asif ud-Doula).
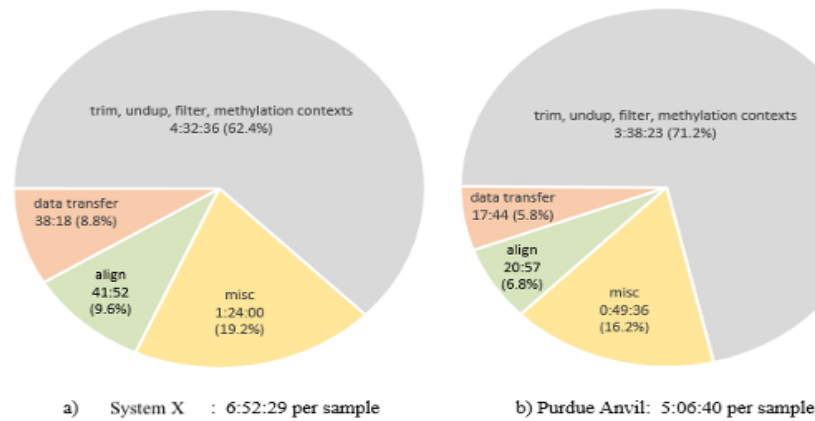

Figure 5: Average end-to-end processing time and percentage of overall processing time per WGBS sample on Anvil versus another XSEDE capacity system. (Images courtesy of Richard Wilton in [19]).

activated exchanges are not yet well understood. This group had previously developed Gaussian accelerated molecular dynamics computational methods to accelerate biomolecular simulations by orders of magnitude. Anvil GPUs enabled the group to carry out long-timescale (2000 ns) simulations (Figure 6) with greater speeds to capture the spontaneous activation of the G protein. In their simulations, Anvil achieved ~ 1.48 speedup when compared to other capacity XSEDE systems due to the A100 GPUs and the filesystem I/O speeds.

6.2.4 Biophysics. This early user group sought to apply Anvil to the task of carrying out all-atom molecular dynamics (MD) simulations to calculate the diffusivity and permeability coefficients for small molecules across a synthetic β-carboxysome shell. Carboxysome proteins form a selective permeable protein shell encapsulating photosynthesis enzymes and play a critical role in carbon fixation for plant energy production. Calculations of permeability coefficients across a synthetic carboxysome shell enable the researchers to analyze CO_2 concentrating mechanisms that are posited to be operating in photosynthetic bacteria. The A100 GPUs

and NVLINK interconnects between the four GPUs on one Anvil node provides a 2-3x performance boost for their MD simulations when compared to the V100 GPUs available on their campus HPC systems (Figure 7).

6.3 User Feedback

In late December 2021 the Anvil team carried out an anonymous online survey of the early users. Users were asked seven Likert scale questions and one additional question on how Anvil has helped their research. 18 responses were received out of 40 active users on Anvil (estimated based on the number of users active on the Anvil support Slack channel). The responses to the free form question overwhelmingly cited the Anvil A100 GPUs and the filesystem (GPFS) I/O performance as being responsible for improved performance results. Similarly, the response was overwhelmingly positive on the seven Likert scale questions. All 18 respondents reported access and login to Anvil as being either extremely or somewhat easy, 14 respondents were able to run their first successful job within a day, 17 respondents found all the necessary compilers, libraries,
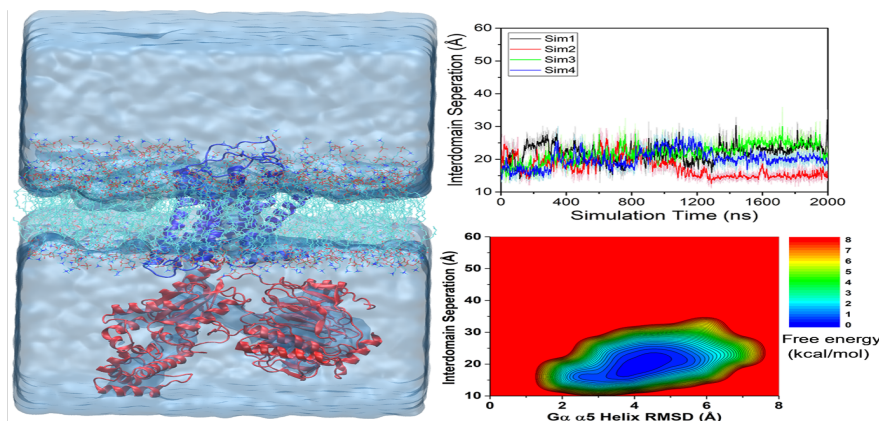
Figure 6: 2000 ns GaMD simulations captured spontaneous activation of the G protein by the GPCR. (Images courtesy of Yinglong Miao).
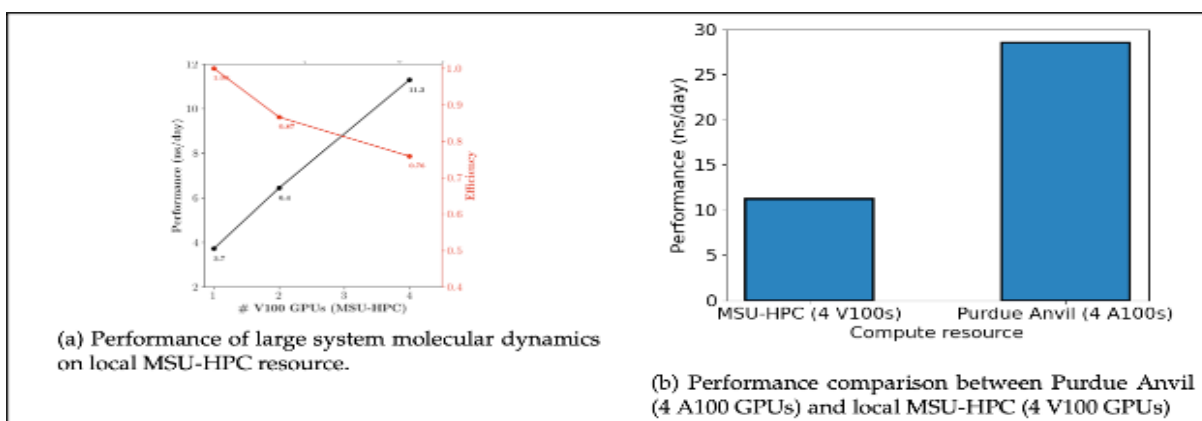
Figure 7: Performance comparison between A100 (Anvil) and V100 (campus resource) GPUs for MD simulations. (Images courtesy of Joshua Vermaas).

and applications available on Anvil, and 13 respondents reported achieving better performance on Anvil with none reporting worse performance.

7 SUMMARY

In this paper, we have described the experiences and outcomes of Purdue University's acquisition and deployment of Anvil - providing advanced CI capabilities and services to support the full range of computational and data-intensive research across all of science and engineering. Following a 56-day early-user operations period where Anvil demonstrated production operation capabilities, with 97% uptime, in support of science and engineering for 29 early user research groups, Anvil has entered production operations on XSEDE.

REFERENCES

[1] G K Andino, M Brazil, M Gribskov, and P Smith. 2017. Advancing the representation of women in HPC at purdue university. In *ACM International Conference Proceeding Series*, Vol. Part F1287. https://doi.org/10.1145/3093338.3104175

[2] Michael E Baldwin, Xiao Zhu, Preston M Smith, Stephen Lien Harrell, Robert Skeel, and Amiya Maji. 2016. Scholar: a campus HPC resource to enable computational literacy. In *Education for High-Performance Computing (EduHPC), 2016 Workshop on*. IEEE, 25–31.

[3] Brendan Burns, Brian Grant, David Oppenheimer, Eric Brewer, and John Wilkes. 2016. Borg, Omega, and Kubernetes. *Queue* 14, 1 (jan 2016), 70–93. https://doi.org/10.1145/2898442.2898444

[4] Cendio AB. [n.d.]. ThinLinc - a remote desktop server from Cendio | ThinLinc by Cendio. ([n. d.]). https://www.cendio.com/thinlinc/what-is-thinlinc

[5] Purdue Research Computing. [n.d.]. ITaP Research Computing - Data Workbench. ([n. d.]). https://www.rcac.purdue.edu/compute/workbench

[6] Felipe da Veiga Leprevost, Björn A Grüning, Saulo Alves Aflitos, Hannes L Röst, Julian Uszkoreit, Harald Barsnes, Marc Vaudel, Pablo Moreno, Laurent Gatto, Jonas Weber, Mingze Bai, Rafael C Jimenez, Timo Sachsenberg, Julianus Pfeuffer, Roberto Vera Alvarez, Johannes Griss, Alexey I Nesvizhskii, and Yasset Perez-Riverol. 2017. BioContainers: an open-source and community-driven framework for software standardization.

Bioinformatics 33, 16 (03 2017), 2580–2582. https://doi.org/10.1093/bioinformatics/btx192 arXiv:https://academic.oup.com/bioinformatics/article-pdf/33/16/2580/25163480/btx192.pdf

[7] Todd Gamblin, Matthew LeGendre, Michael R. Collette, Gregory L. Lee, Adam Moody, Bronis R. de Supinski, and Scott Futral. 2015. The Spack package manager. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis on - SC '15*. ACM Press, New York, New York, USA, 1–12. https://doi.org/10.1145/2807591.2807623

[8] Erik Gough. [n.d.]. ITaP Research Computing - Interactive Scientific Computing on the Anvil Composable Subsystem. https://www.rcac.purdue.edu/training/anvilcomposable

[9] Erik Gough, Sam Weekly, Brian Werts, and Preston Smith. 2021. Enabling Research and Education through the Geddes Composable Platform. (nov 2021). https://doi.org/10.5281/ZENODO.5796289

[10] S Harrell, H A Nam, V Vergara, K Keville, and D Kamalic. 2015. Student Cluster Competition: A Multi-disciplinary Undergraduate HPC Educational Tool. In *EduHPC '15 Proceedings of the Workshop on Education for High-Performance Computing*.

[11] Stephen Lien Harrell, Marisa Brazil, Alex Younts, Daniel T. Dietz, Preston Smith, Erik Gough, Xiao Zhu, and Gladys K. Andino. 2018. Mentoring Undergraduates into Cyber-Facilitator Roles. In *Proceedings of the Practice and Experience on Advanced Research Computing - PEARC '18*. ACM Press, New York, New York, USA, 1–7. https://doi.org/10.1145/3219104.3219138

[12] Dave Hudak, Doug Johnson, Alan Chalker, Jeremy Nicklas, Eric Franz, Trey Dockendorf, and Brian L Mcmichael. 2018. Open OnDemand: A web-based client portal for HPC centers Software • Review • Repository • Archive. (2018). https://doi.org/10.21105/joss.00622

[13] NVIDIA. [n.d.]. NGC Overview :: NVIDIA GPU Cloud Documentation. ([n. d.]). https://docs.nvidia.com/ngc/ngc-overview/index.html

[14] Rancher. [n.d.]. Enterprise Kubernetes Management | Rancher. https://rancher.com/

[15] John Towns. 2020. XSEDE Annual Report for Report Year 4 and Program Plan for Project Year 10. (may 2020). https://www.ideals.illinois.edu/handle/2142/107285

[16] J. Towns, T. Cockerill, M. Dahan, I. Foster, K. Gaither, A. Grimshaw, V. Hazlewood, S. Lathrop, D. Lifka, G. D. Peterson, R. Roskies, J. Scott, and N. Wilkins-Diehr. 2014. XSEDE: Accelerating Scientific Discovery. *Computing in Science & Engineering* 16, 05 (sep 2014), 62–74. https://doi.org/10.1109/MCSE.2014.80

[17] Sam Weekly, Zoey Mertes, and Alex Younts. 2021. Rapid Prototype for Shifting HPC to the Cloud. *2021 IEEE 17th International Conference on eScience (eScience)* (sep 2021), 262–263. https://doi.org/10.1109/ESCIENCE51609.2021.00051

[18] Sage A Weil, Scott A Brandt, Ethan L Miller, Darrell D E Long, and Carlos Maltzahn. 2006. Ceph: A Scalable, High-Performance Distributed File System. In *Proceedings of the 7th Symposium on Operating Systems Design and Implementation (OSDI '06)*. USENIX Association, USA, 307–320.

[19] Richard Wilton and Alexander S Szalay. 2022. Performance optimization in DNA short-read alignment. *Bioinformatics* 38, 8 (02 2022), 2081–2087. https://doi.org/10.1093/bioinformatics/btac066 arXiv:https://academic.oup.com/bioinformatics/article-pdf/38/8/2081/43369967/btac066.pdf

[20] Tsai-Wei Wu, Stephen Lien Harrell, Geoffrey Lentner, Alex Younts, Sam Weekly, Zoey Mertes, Amiya Maji, Preston Smith, and Xiao Zhu. 2021. Defining Performance of Scientific Application Workloads on the AMD Milan Platform. In *Practice and Experience in Advanced Research Computing (PEARC '21)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3437359.3465596