Accelerating Advanced Light Source Science Through Multi-Facility HPC Workflows

David Abramov*
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
ALS
Berkeley, California, USA
dabramov@lbl.gov

Samuel Welborn*
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
NERSC
Berkeley, California, USA
swelborn@lbl.gov

Ryan Chard
Argonne National Laboratory (ANL)
Chicago, Illinois, USA
rchard@anl.gov

Kuldeep Chawla
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
LBL IT
Berkeley, California, USA
kchawla@lbl.gov

Xiaoya Chong
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
ALS
Berkeley, California, USA
xchong@lbl.gov

Elizabeth Clark
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
ALS
Berkeley, California, USA
elizabethclark@lbl.gov

Bjoern Enders
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
NERSC
Berkeley, California, USA
benders@lbl.gov

Alexander Hexemer
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
ALS
Berkeley, California, USA
ahexemer@lbl.gov

Jason Jed
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
ALS
Berkeley, California, USA
jjed@lbl.gov

Wiebke Koepp
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
ALS
Berkeley, California, USA
wkoepp@lbl.gov

Harinarayan Krishnan
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
CAMERA
Berkeley, California, USA
hkrishnan@lbl.gov

Seij De Leon
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
ALS
Berkeley, California, USA
seijdeleon@lbl.gov

Dilworth Parkinson
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
ALS
Berkeley, California, USA
dyparkinson@lbl.gov

David Perlmutter
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
CAMERA
Berkeley, California, USA
dperl@lbl.gov

Raja Vyshnavi Sriramoju
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, California, USA
ALS
Berkeley, California, USA
rvsriramoju@lbl.gov

Thomas Uram
Argonne National Laboratory (ANL)
Argonne, Illinois, USA
ALCF
Argonne, Illinois, USA
turam@anl.gov

Lee Lisheng Yang
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, USA
ALS
Berkeley, California, USA
llyang@lbl.gov

Dylan McReynolds
Lawrence Berkeley National Laboratory (LBNL)
Berkeley, USA
ALS
Berkeley, California, USA
dmcreynolds@lbl.gov

Abstract

Synchrotron light sources support a wide array of techniques to investigate materials, often producing complex, high-volume data that challenge traditional workflows. At the Advanced Light Source (ALS), we developed infrastructure to move microtomography data over ESnet to ALCF and NERSC, where CPU- and GPU-based algorithms generate 3D reconstructed volumes of experimental samples. We employ two data movement and reconstruction models: real-time processing as data streams directly to NERSC compute nodes, and automated file transfer to NERSC and ALCF file systems. The streaming pipeline provides users with feedback in under ten seconds, while the file-based workflow produces high-quality reconstructions suitable for deeper analysis in 20-30 minutes. This infrastructure enables users to utilize HPC resources without direct access to backend systems. We plan to extend this architecture to more endstations, supporting our beamline scientists and users.

CCS Concepts

• Applied computing → Physical sciences and engineering; **• Software and its engineering**; **• Human-centered computing → Visualization**; **• Computer systems organization → Cloud computing**;

Keywords

synchrotron, tomography, streaming, HPC workflows, automation

ACM Reference Format:
David Abramov, Samuel Welborn, Ryan Chard, Kuldeep Chawla, Xiaoya Chong, Elizabeth Clark, Bjoern Enders, Alexander Hexemer, Jason Jed, Wiebke Koepp, Harinarayan Krishnan, Seij De Leon, Dilworth Parkinson, David Perlmutter, Raja Vyshnavi Sriramoju, Thomas Uram, Lee Lisheng Yang, and Dylan McReynolds. 2025. Accelerating Advanced Light Source Science Through Multi-Facility HPC Workflows. In *Workshops of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC Workshops '25), November 16–21, 2025, St Louis, MO, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3731599.3767590

1 Introduction

The Advanced Light Source (ALS) is a Department of Energy (DOE) third-generation X-ray synchrotron serving thousands of users annually across materials science, biology, chemistry, and physics. Historically, experiments produced manageable data volumes that could be analyzed on workstations local to the beamline. However, modern, high-bitrate detectors and improvements in X-ray beam brightness and coherence is shifting this paradigm. Data throughput is expected to increase by orders of magnitude with the next upgrade to ALS (ALS-U), with projected facility-wide volumes reaching multiple petabytes per day [1].

Aligning with emerging DOE initiatives [19], we addressed these challenges by designing and deploying a novel infrastructure that integrates the ALS microtomography beamline (8.3.2) [15] with two high-performance computing (HPC) facilities: the National Energy Research Scientific Computing Center (NERSC) and Argonne Leadership Computing Facility (ALCF). This framework automates the movement of raw data from the beamline acquisition server to HPC centers, processes the data with CPU- and GPU-based tomographic reconstruction code, and serves the results back to the user. We achieve both rapid turnaround time (<10s) for immediate feedback by processing detector data streams synchronously with acquisition on NERSC compute nodes, and high-quality reconstructions by performing longer-running file-based processing on both ALCF and NERSC compute nodes. We serve both types of reconstructions back to the user for seamless visualization and interaction.

In this manuscript, we present an end-to-end production implementation of our microtomography beamline computing infrastructure that integrates with existing control software, and supports automated, low-latency reconstruction workflows for scientific users. We first describe the background information on data challenges at light sources and user facilities in **Section 2**. Then, we walk through the user experience of microtomography beamline users in **Section 3**. After this, we discuss in detail the overall design and implementation of the system in **Section 4**, followed by an evaluation our infrastructure from qualitative and quantitative perspectives in **Section 5**. Finally, we share future directions in **Section 6** and closing thoughts in **Section 7**.

2 Background and Related Work

2.1 Data Challenges (and Solutions) at DOE Light Sources and User Facilities

Synchrotron facilities enable researchers to probe matter at nanometer resolution and femtosecond timescales across diverse experimental conditions. Common techniques, including tomography (3D structural imaging), ptychography (high-resolution phase contrast), ARPES (electronic band structure), and various scattering methods, generate multidimensional datasets requiring coordination between beamline control systems, data storage, and analysis pipelines [4]. As modern detectors increase in speed and resolution, facilities globally face unsustainable data rates and fragmented workflows [13]. This "data deluge" affects everything from local storage to curation and real-time feedback [33].

The ALS is preparing for the ALS-U upgrade [26], which will transform it into a fourth-generation, diffraction-limited storage ring. This transformation will increase data rates through enhanced synchrotron brightness [1]. In parallel, advances such as ultrafast detectors and autonomously-driven experiments are poised to further accelerate scientific discovery [16, 36]. Similar upgrades at other user facilities demonstrate the scale of this challenge: the Advanced Photon Source (APS) developed standardized HPC workflows [24] and LCLS-II built scalable infrastructure for live interactive analysis [6, 28]. Cloud-based orchestration frameworks have been developed for managing terabyte scale materials workflows in tomography [37]. Recent work at the National Center for Electron Microscopy (NCEM) shows how streaming to HPC over high-speed networks significantly improves data turnaround and real-time experiment steering [34].

*Both authors contributed equally to this research.

ACM ISBN 979-8-4007-1871-7/25/11
https://doi.org/10.1145/3731599.3767590

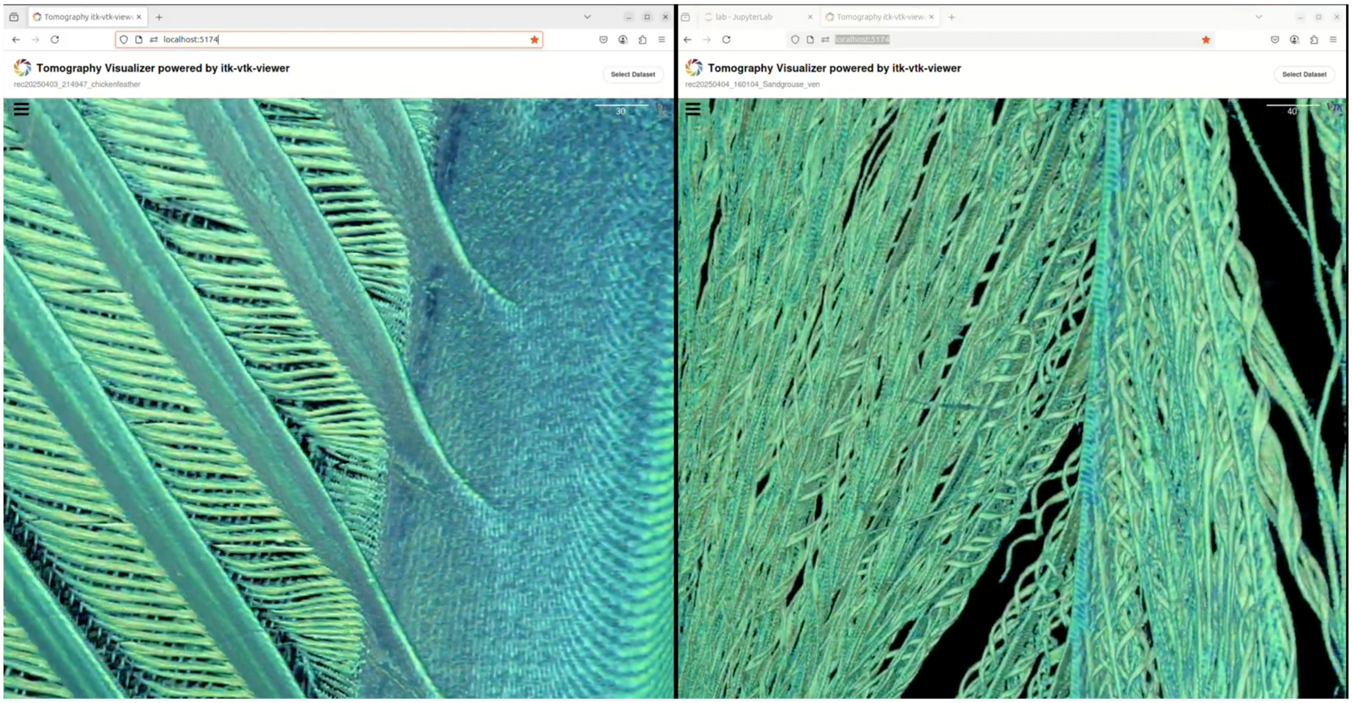

Figure 1: Comparison between a chicken (left) and sandgrouse (right) feather tomography reconstructions. HPC-augmented workflows help users at the microtomography beamline quickly inspect material characteristics, such as morphological differences between species as discussed further in Section 5.1.1.

2.2 DOE Computing Infrastructure

The DOE Public Access Plan [29] outlines that DOE-funded data should follow FAIR principles (Findable, Accessible, Interoperable, Reusable) and data management best-practices [35], including persistent identifiers, rich metadata, and immediate availability. As outlined in the "AI@ALS Workshop Report" [23], meeting FAIR data requirements is essential for supporting advanced artificial intelligence and end-to-end data analysis pipelines, which necessitates investment into data infrastructure, collaboration with other institutions, and fostering a culture of continuous improvement.

Addressing these data challenges, we leverage the DOE's Advanced Scientific Computing Research (ASCR) infrastructure, including high-performance computing facilities (NERSC, ALCF), the Energy Sciences Network (ESnet), and the Integrated Research Infrastructure (IRI) initiative [19]. Through IRI, we meet regularly with colleagues from across light sources and ASCR facilities to find commonalities and share lessons learned. Recent advances demonstrate the potential: streaming to supercomputers for real-time processing [30], federated ptychographic reconstructions [5], and optimized ML inference workflows [14]. However, deployment requires addressing facility-specific challenges such as priority scheduling for time-critical workflows, and managing distinct authentication and containerization across diverse HPC architectures.

3 User Experience

The effectiveness of data infrastructure in a user facility such as the ALS ultimately hinges on the experience of its users, scientists, engineers, and collaborators who must interact with beamline control systems, reconstruction pipelines, and analysis environments, often under tight experimental schedules. **Table 1** outlines a set of three beamline user archetypes we identified to guide our infrastructure, each with distinct needs and levels of interaction. This user-centric approach ensures that design decisions, stakeholder communication, and strategic planning reflect the needs of the people who depend on the system daily. See **Section 5.1** for user feedback.

Table 1: Beamline User Archetypes

User Type	Description
Visiting User	Short, on-site scheduled beamtime; requires remote data access; focused on rapid data acquisition under constrained timeframes; thousands of annual users (novices and experts); utilizes the tools in **Figure 2**.
Staff Beamline Scientist	Endstation expert (hardware, software, analysis); provides guidance to users; ensures experimental quality and system uptime; 1-2 per beamline.
Software Engineer	Develops and maintains scalable infrastructure, compute and visualization services; software support for the tools and flows in **Figure 2**, and **Figure 3**.

3.1 Microtomography Workflow Use Case

We highlight a representative use case from the ALS microtomography beamline [15], where the integration of HPC resources enables rapid processing for fast feedback during an experiment. Almost immediately, 2D reconstructed previews are visible to the user. Within a few more minutes, users can interact with their data in a browser-based 3D volume visualization. **Figure 2** shows an overview of the tools users interact with at this beamline.

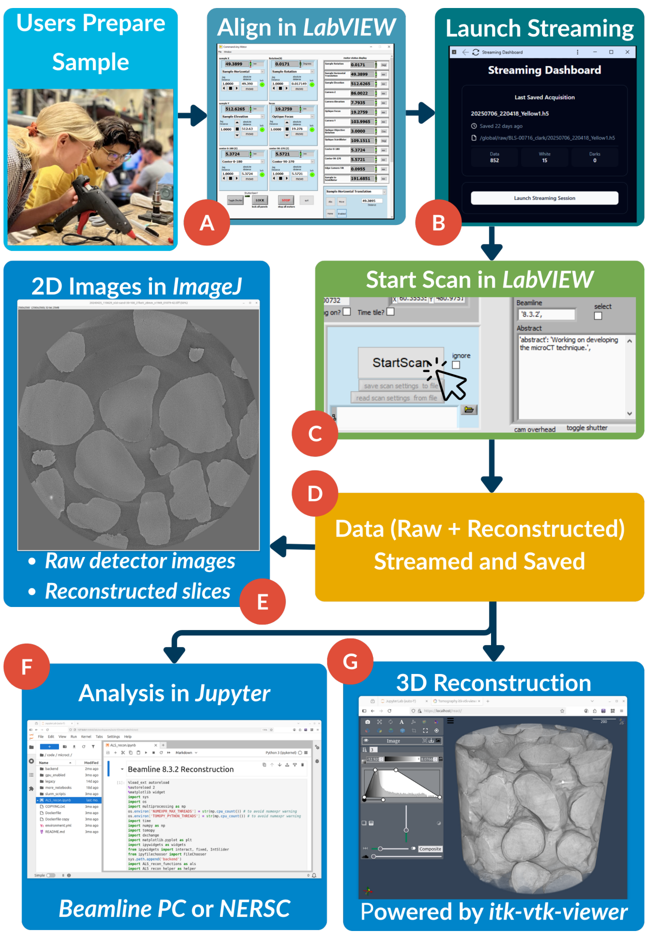

Figure 2: Workflow interfaces at the microtomography beamline. Users align samples in LabVIEW, launch the streaming service via a web app, and start scans. Orthogonal slices of the reconstructed volume appear in ImageJ within 10 seconds of acquisition completion. After reconstruction, users explore persistent data via Jupyter, view TIFF slices in ImageJ, and inspect volumes using an itk-vtk-viewer web app.

Workflow Overview. Users engage with beamline controls software and web interfaces, with no direct access to backend computing infrastructure required. They begin by preparing and mounting their sample (**Figure 2**), and aligning it to the beam using control software (**Figure 2A**). Users have the option of launching a NERSC streaming service using the web application shown in **Figure 2B**. Upon starting a scan (**Figure 2C**), a 180° image sequence is saved as an HDF5 file with embedded metadata.

Our framework contains two main workflow branches that run in parallel: (i) the streaming branch and (ii) the file transfer branch. After completion of an acquisition, the streaming service at NERSC reconstructs the full dataset and sends a three-slice preview back to the user in less than 10 seconds (**Figure 2D** and **E**). On the file transfer branch, HPC workflows are triggered at both NERSC and ALCF as soon as the acquisition has finished saving to the local beamline server. This branch takes longer to complete (20-30 minutes), but produces higher quality reconstructions owing to the preprocessing and iterative algorithms used. In addition, metadata is captured automatically, outputs are persistently registered, and remote collaborators can participate with VPN access.

Reconstructed 3D volumes from the file transfer branch become available to the user in a web-based viewer powered by *itk-vtk-viewer* and *Bluesky Tiled* (**Figure 2G**), which we extended to support immersive visualization using WebXR, and demonstrated on Meta Quest 3 headsets. Users can explore their metadata via SciCat [20], transfer data using Globus, and initiate further analysis in JupyterLab (**Figure 2F**), as well as segmentation and feature extraction in dedicated applications such as Dragonfly [7] and the MLExchange segmentation app [14].

4 Design and Implementation

We designed our system to address four primary challenges. First, *beamline infrastructure is heterogeneous.* ALS uses a mix of LabVIEW and EPICS control systems, while DOE compute centers expose Slurm, PBS, or Globus Compute interfaces. Second, *data volumes are substantial.* Each 3-minute scan usually produces 20–30 GB of raw images, and reconstruction produces an additional 40–60 GB of data. Since storing the data on multiple intermediate file systems introduces feedback latency, we implement dual-path processing: Globus-based transfers for file-based (batch) workflows for high-quality reconstructions, and synchronous processing on EPICS-based data streams for rapid feedback. Third, *local beamline computing resources are limited.* Thankfully, we can leverage DOE HPC resources to execute remote jobs, however, this still requires facility-specific implementations. Fourth, *the ALS supports dozens of beamlines.* While initially deployed at the microtomography beamline, we aim to generalize our work to other endstations with data movement, launching technique-specific analysis codes on HPC, and metadata management. The infrastructure we present is enabling a pilot deployment on a second beamline, forming a template for additional planned rollouts.

4.1 Technology Overview

We integrate an open-source software stack in our GitHub repository **splash_flows** [10]. Workflows are orchestrated by a **Prefect** server running in a virtual machine (**VM**), which directs jobs to Prefect Workers in the same VM. When files are written to disk as an HDF5 file, we call Prefect to copy the data from the ALS to HPC centers using **Globus Transfer** [3, 9]. We have demonstrated the monitoring of Globus data transfer bandwidth with **Grafana**. In parallel, projection image streams are forwarded to NERSC using **pvaPy** [30], reconstructed with **streamtomocupy** [22], and results are streamed back to the ALS via **ZeroMQ** [12]. A **React**-based single-page web application at the beamline facilitates initiation and monitoring of the NERSC streaming service through the Superfacility API (**SFAPI**) [8]. HPC compute for the file-based workflow is split between **NERSC**—Slurm jobs are launched through the **SFAPI** or in an interactive **JupyterLab** session—and remote file-based reconstruction at **ALCF** via **Globus Compute**. Both HPC file-based workflows utilize **TomoPy** [11] for reconstruction. Software environments are packaged in **Docker**/**Podman** containers configured with Conda environments, often inside lightweight Ubuntu VMs. Users can inspect reconstruction data with **itk-vtk-viewer** [17, 18, 25], **Bluesky Tiled** (a data access service) [2], the MLExchange segmentation app [14], or in VR. Another **React**-based single-page web application provides the web interface for the volume viewer and data selection. **SciCat** [20] records metadata, and long-term datasets migrate to the high-performance storage system (**HPSS**) tape at NERSC.

4.2 Operational Layers

As illustrated in **Figure 3**, the architecture separates concerns across five cooperating layers.

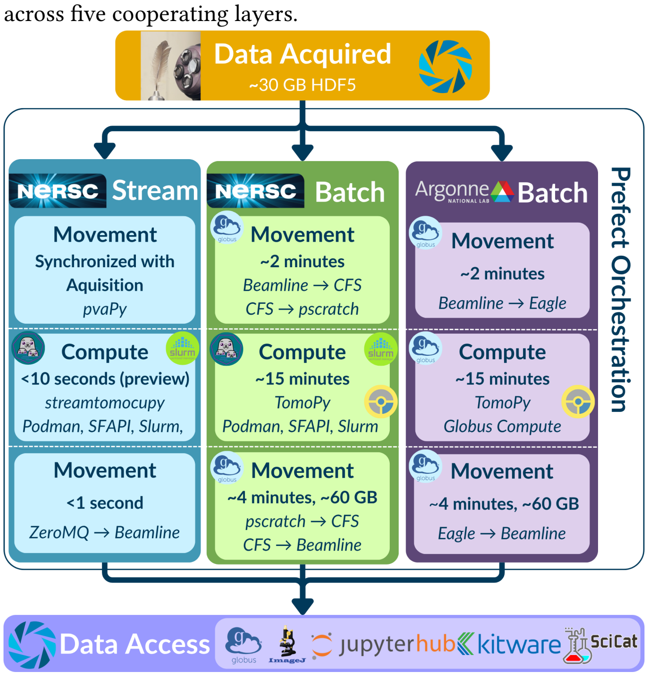

Figure 3: Overview of the operational layers in our infrastructure: *Acquisition*, *Orchestration*, *Movement*, *Compute*, and *Access*. Each scan usually generates around 30 GB of raw data, but this can vary depending on the scan settings. The NERSC streaming branch provides immediate visual feedback to the user, while the two file-based workflows (NERSC and ALCF) generate persistent data products for further analysis.

4.2.1 *Acquisition Layer.*

The microtomography beamline detector uses an EPICS driver for data acquisition. To save acquisitions, we employ two pvaPy-based systemd services on beamline's local storage server: (1) a process variable access (PVA) channel mirror server, which republishes data from the detector's input/output controller (IOC), and (2) a file-writing service. [30] The PVA mirror server publishes detector frames data for both the file-writing service and the optional streaming reconstruction service running at NERSC. For each frame, the file-writing service first validates its metadata, and then uses the metadata to write into an HDF5 file.

4.2.2 *Orchestration Layer.*

A call to Prefect from the file-writing service initiates a set of flows once an acquisition is written to disk.

Prefect Flows. The **new_file_832** flow orchestrates data movement between beamline servers, as well as metadata ingestion into SciCat. The file-based reconstruction flows, **alcf_recon_flow** and **nersc_recon_flow**, copy data to HPC using Globus Transfer, launch TomoPy reconstruction in CPU nodes, and generate a stack of TIFF images and a multi-scale reconstructed volume (Zarr format). Once complete, these HPC flows coordinate data movement back to the beamline. We encapsulate these variations with adapter classes, ensuring location-transparent orchestration. To launch and monitor our streaming service at NERSC via SFAPI, users initiate a flow using our simple web frontend (shown in **Figure 2B**). Additionally, scheduled *pruning flows* prevent storage saturation. Prefect flows can be retried in the user interface in case of errors.

Prefect Workers. Prefect workers execute flows in isolated containers with carefully tuned limits: tuned concurrency for scan detection tasks, but lower concurrency for HPC job submission to prevent queue conflicts. Workflows are designed as a series of subflows and tasks, implementing idempotent semantics that support safe retries of specific steps in case of failure.

Container Deployment. We deploy services using Docker containers built by GitHub Actions runners, tagged with version numbers. This continuous integration/continuous deployment (CI/CD) pipeline enables rapid iteration while maintaining stability during beamtime. We freeze container versions during experiments and update only during scheduled maintenance windows.

Virtual Machine. Our production Prefect server and related services runs on a dedicated Ubuntu 20.04.6 virtual machine within the ALS-managed VMware environment. It is provisioned with an 8-core Intel® Xeon® Gold 5120 CPU (2.20GHz, AVX-512), 8GiB RAM, and a 2GiB swap partition. The local storage stack comprises a 78GiB logical volume and a 20TiB NFS-mounted high-throughput volume, used for data staging and accessed by Prefect workers. Networking is provisioned via a 10 Gbps full-duplex Twisted Pair VMXNET3 virtual NIC, with VLAN-based segmentation.

4.2.3 *Data Movement Layer.*

Our system transfers data to the HPC centers with two parallel mechanisms: streaming using EPICS and pvaPy, and file transfer using Globus.

Streaming Pipeline. The streaming pipeline addresses the critical need for rapid experiment feedback. The advantage here is that we skip the intermediate movement steps of raw data to both the beamline server and the shared file systems at HPC facilities. When the streaming service is running at NERSC, it connects to the beamline's PVA mirror server and receives frames as they are acquired synchronously with the file-writer service (**Section 4.2.1**). This service stores the frames in an in-memory cache until the acquisition is complete, and then performs a back projection using streamtomocupy with all four GPUs on a NERSC GPU node [21, 22].

File-Based Pipeline. Once the file-writing service completes an acquisition, it triggers the file-based Prefect Flow on the raw HDF5 dataset. Each file moves from the acquisition system to a user-accessible beamline data server. From here, we move data to the NERSC Community Filesystem (CFS) and the Eagle filesystem at ALCF. After reconstructions are processed, the data is copied back to the beamline data server. Each of these movements is coordinated using the Globus Transfer Python SDK [3, 9] for streamlined authentication. We enable checksum verification to ensure data integrity when moving files and folders between locations. Within the NERSC file-based reconstruction Slurm job, we copy data using bash commands from CFS to Perlmutter Scratch (pscratch) for improved I/O performance. Transfer flows to and from HPSS for long-term archival are also handled through Slurm and SFAPI. The pruning flows built into the orchestration layer provide scheduled file-system clean up across all levels to avoid storage saturation.

4.2.4 ***Compute Abstraction Layer.*** Our infrastructure provides facility-specific compute adapters that handle each system's specific authentication, job submission, and result collection protocols. In addition to these adapters, users can manually perform reconstructions and further analysis in JupyterHub on a beamline PC or via NERSC.

The NERSC adapter uses the Superfacility API (SFAPI) to submit Slurm jobs, supporting both streaming and file-based reconstruction services. We submit reconstruction jobs to NERSC via SFAPI using ALS's collaboration account [8]. These jobs are scheduled with the realtime quality-of-service (QOS), which provides prioritized job scheduling. Jobs execute within podman-hpc [27] containers that mount the reconstruction scripts and data volumes. For the file-based reconstruction, we request exclusive access to a full CPU node with 128 cores for at least a 15-minute window.

Our ALCF adapter implements reconstruction using a serverless approach via Globus Compute, which uses a pilot-job model to maintain compute nodes that can be reused when they are available, as well as a demand queue on Polaris to reduce queue wait times. Remote Python functions are executed on the Polaris HPC cluster through pre-configured endpoints, accessing the Eagle filesystem and invoking reconstruction scripts as subprocess calls. This function-as-a-service approach enabled by Globus Compute, combined with the use of a demand queue, provides immediate execution without the overhead of traditional batch scheduling.

4.2.5 ***Access Layer.*** Results are made accessible to users both at the beamline and remotely. At the microtomography beamline, raw and reconstructed datasets are stored on a data server with a Globus transfer endpoint [3, 9], which is mounted to the beamline computers. At NERSC, data is stored on CFS, and user data is made available in JupyterHub. Metadata for each scan is searchable in SciCat. Additionally, we provide visualization tools for users to inspect 2D reconstructed images in ImageJ, and 3D reconstructed volumes are served via Bluesky Tiled [2] to a web app powered by itk-vtk-viewer [17, 18, 25], which supports real-time and immersive viewing via a desktop web browser and in VR.

4.3 Data Lifecycle

Each scan follows a complete pipeline from acquisition through reconstruction. The system supports horizontal scaling via multiple Prefect agent pools and configurable compute resources, though current deployments use fixed allocations. Under typical operation, the system processes peak data rates of one scan every 3-5 minutes (12-20 scans/hour), with daily volumes ranging from 0.5-5 TB depending on the experiment. Raw file sizes range from a few MB to hundreds of GB, but typical scientific scans are between 20-30 GB. Data is tiered through distributed network storage for fast writing and user-access, longer-term storage on the NERSC Community Filesystem (CFS), and archival on HPSS. Storage is managed through automated age-based pruning flows, with retention periods optimized for each tier (local servers: days to weeks, CFS: months to years, HPSS: indefinite long-term archive).

5 Evaluation

We evaluate our integrated tomography pipeline along two complementary axes: *usability*, as reported by end users (qualitative), and *infrastructure performance*, based on timing benchmarks across beamline and HPC resources (quantitative). Our findings demonstrate that the infrastructure meaningfully reduces experimental friction for each user archetype we identified in **Table 1**, and improves the throughput and reproducibility of scientific data analysis.

5.1 User Feedback

A decade-long beamline user highlighted the dramatic improvement: "When I started, it took 45 minutes just to save a scan, then another hour to get back a single reconstruction slice—not even the full volume. Now I get complete 3D volumes in minutes." This 100× speedup fundamentally changes how experiments are conducted, enabling previously impossible real-time decision-making.

5.1.1 Case Study 1: Feather Morphology Comparison. We compared the microstructure of chicken and sandgrouse feathers (). The sandgrouse has evolved specialized coiled barbule structures that store water, an adaptation for desert survival absent in chicken feathers. Our pipeline enabled rapid sample exchange and side-by-side volumetric comparison, immediately revealing these structural differences. This workflow—mount, scan, reconstruct, compare—now takes 20 minutes instead of hours, accelerating studies.

5.1.2 Case Study 2: Fracking Proppant Analysis: A Retrospective. Beamline scientists often need to communicate complex results, and our system makes historical data easy to reprocess and share with other stakeholders. We reanalyzed a 2020 micro-CT dataset of fracking proppant [31, 32], reconstructing and segmenting the raw data using our infrastructure. The 3D volume was textured in Blender and exported for virtual reality (VR). During a recent tour, visitors explored the model in a Meta Quest 3 headset, highlighting how our infrastructure supports scientific communication.

5.1.3 Case Study 3: Software Support. Software engineers are responsible for connecting the tooling, computing, and persistence required for daily operation. The orchestration layer powered by Prefect provides a friendly user interface, making it easy to identify flow status, timelines, and errors for each step. Logs are stored in a database, made available directly in the browser, and update in real-time. In addition to debugging, the Prefect API allows for extracting flow statistics and observing run success rate (**Table 2**).

5.2 Performance Metrics

We analyzed 100 recent successful microtomography scans in the file-transfer workflow branch, encapsulating the time for initial data staging and metadata ingestion (*new_file_832*) and the HPC file-based workflows (*alcf_recon_flow*, *nersc_recon_flow*). We queried the Prefect server API, extracted and aggregated completion times as summarized in **Table 2**[1]. The range in processing time is explainable by the possible files sizes. Cropped test scans produce small files of only a few MB, whereas uncropped scans at high angular resolution are much larger, possibly greater than 30 GB.

Table 2: Summary statistics of the last 100 successful file-based Prefect flow runs in production for user operations. Durations shown in seconds.

Flow	N	Mean ± SD	Med.	Range
new_file_832	100	120 ± 171	56	[30, 676]
nersc_recon_flow	100	1525 ± 464	1665	[354, 2351]
alcf_recon_flow	100	1151 ± 246	1114	[710, 1965]

For the fast turnaround streaming workflow branch, the user can preview their reconstructed data within 10 seconds of acquisition completion. For example, a raw dataset with 1969 16-bit projection images of size 2160×2560 (~20 GB), takes 7–8 seconds to reconstruct, with a reconstructed volume size of 2160 × 2560 × 2560 32-bit (~50 GB). Sending the preview slices back to ALS takes <1 second.

Key quantitative outcomes:

- **>100× improvement** in time-to-insight compared to historical workflows with streaming reconstruction previews achieved in <10 seconds after acquisition completes;
- **Consistent performance:** median file-based reconstruction times in 20-30 minutes (optimized for image quality).

5.3 Strengths and Limitations

We developed infrastructure that directly serves the needs of beamline users (**Table 1**) by coordinating data movement, processing, and visualization. We addressed multi-institutional challenges through regular IRI meetings [19], submitting tickets to HPC help-desks, and writing facility-specific job submission implementations that allows us to run the same analysis code across facilities. Users gain the benefits of HPC-powered analysis without interacting with the systems directly, and leave with derived data. By integrating multiple computing centers, we are increasing our current fault-tolerance and are preparing for future data needs. Our containerized services, versioned workflows, and persistent metadata ensure reproducibility and scalability from laptop development to distributed systems. Staff have observability into the underlying systems, and we aim for this architecture to be extensible across beamlines.

Current limitations include storage overhead and maintenance complexity. While we are capturing *instrument* metadata from each scan into SciCat, a major limitation is the absence of standardized *sample* metadata capture, including provenance, preparation methods, in situ conditions, and material classifications. This gap impedes experiment contextualization and future AI-driven analysis that require rich semantic information.

Production lessons learned include: maintaining strict staging and production separation, automated health monitoring every 12-24 hours, and version-controlled deployments. In one incident, a burst of concurrent Globus Transfer "prune" requests hit a permission denied error, leaving a slew of jobs hanging and saturating the queue. To avoid issues like these, we refactored our flows to fail early, and try to automatically cancel jobs on remote systems. Detailed documentation is essential, especially in cases when the person who created or maintains part of the system is unavailable.

[1] https://github.com/als-computing/splash_flows/tree/XLOOP_SC25_Metrics

6 Discussion and Future Directions

Our implementation demonstrates an end-to-end pipeline that ingests raw projection data, orchestrates transfers and reconstructions across multiple HPC sites, and delivers interactive visualization to users. By combining a workflow orchestration layer, robust data movement, and containerized compute environments, we achieved a resilient and extensible system. We identify two impactful directions for future development.

Dynamic and Real-Time Analysis. Leveraging quick streaming reconstructions, we can explore supporting time-resolved experiments by extending our workflow to handle 4D datasets as sequences of time-stamped volumes. Furthermore, optimizing the multi-resolution volume conversion step (Zarr files), rendering 3D volumes on HPC, and implementing WebRTC streaming to VR headsets would better enable immersive experiment steering.

Expanded Compute Resources. Beyond NERSC and ALCF, integration with commercial clouds (AWS, Google) and other DOE facilities (OLCF) would provide additional capacity and specialized hardware. As facilities like NERSC upgrade to next-generation architectures, our containerized approach ensures portability. As more beamlines adopt streaming, the issue shifts from a scheduling to an economic-policy challenge. At scale, compute could be reserved for each beamline to prevent resource contention.

7 Conclusion

We presented an operational HPC-integrated workflow that accelerates science at ALS through automated data movement, synchronous reconstruction, and interactive visualization tools. Deployed at the microtomography beamline, the system processes 0.5-5 TB daily while providing quick visual feedback to users. Key innovations include: scalable framework that supports multi-facility HPC orchestration; on-demand reconstructions via high-speed streaming and high-quality file-based workflows; users harness the computing power of HPC with minimal interaction. As synchrotron facilities worldwide undergo fourth-generation upgrades, this blueprint for beamline-HPC integration becomes increasingly critical. Our open-source implementation [10] provides both a practical solution for current operations and a foundation for future data challenges.

Acknowledgments

We thank Namyi Ha (Georgia Institute of Technology) for providing the feather datasets, Marco Voltolini and Jonathan Ajo-Franklin for the shale-proppant datasets, Christine Simpson for their help with Globus Compute on ALCF, and Viktor Nikitin and Alan Kastengren at Argonne National Lab for their work on streaming. This research used resources of the Advanced Light Source, which is a DOE Office of Science User Facility under contract no. DE-AC02-05CH11231. An award for computer time was provided by the U.S. Department of Energy's (DOE) Innovative and Novel Computational Impact on Theory and Experiment (INCITE) Program. This research used resources from the Argonne Leadership Computing

Facility, a U.S. DOE Office of Science user facility at Argonne National Laboratory, which is supported by the Office of Science of the U.S. DOE under Contract No. DE-AC02-06CH11357. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a Department of Energy User Facility using NERSC award ERCAP0031436 and the NERSC Science Acceleration Program (NESAP).

References

[1] Advanced Light Source. 2024. *Advanced Light Source 5-Year Strategic Plan: FY2024–FY2029*. Technical Report. Lawrence Berkeley National Laboratory. https://als.lbl.gov/about/als-strategic-plan-2024-2029/ Accessed: 2025-07-01.

[2] Daniel Allan, Thomas Caswell, Stuart Campbell, and Maksim Rakitin. 2019. Bluesky's ahead: A multi-facility collaboration for an a la carte software project for data acquisition and management. *Synchrotron Radiation News* 32, 3 (2019), 19–22.

[3] Bryce Allen, John Bresnahan, Lisa Childers, Ian Foster, Gopi Kandaswamy, Raj Kettimuthu, Jack Kordas, Mike Link, Stuart Martin, Karl Pickett, et al. 2012. Software as a service for data scientists. *Commun. ACM* 55, 2 (2012), 81–88.

[4] Soren Bear, Dula Parkinson, Lisa Gerhardt, Mel Sibony, and Bjoern Enders. 2021. Als share, a lightweight data sharing service for a synchrotron radiation facility. In *2021 3rd Annual Workshop on Extreme-scale Experiment-in-the-Loop Computing (XLOOP)*. IEEE, 1–7.

[5] Tekin Bicer, Xiaodong Yu, Daniel J Ching, Ryan Chard, Mathew J Cherukara, Bogdan Nicolae, Rajkumar Kettimuthu, and Ian T Foster. 2021. High-performance ptychographic reconstruction with federated facilities. In *Smoky Mountains Computational Sciences and Engineering Conference*. Springer, 173–189.

[6] Johannes P Blaschke, Felix Wittwer, Bjoern Enders, and Debbie Bard. 2023. How a Lightsource Uses a supercomputer for live interactive analysis of large data sets: Perspectives on the NERSC-LCLS superfacility. *Synchrotron Radiation News* 36, 4 (2023), 10–16.

[7] Comet Technologies Canada Inc. 2024. *Dragonfly*. Montreal, Canada. https://dragonfly.comet.tech/

[8] Bjoern Enders, Debbie Bard, Cory Snavely, Lisa Gerhardt, Jason Lee, Becci Totzke, Katie Antypas, Suren Byna, Ravi Cheema, Shreyas Cholia, et al. 2020. Cross-facility science with the superfacility project at lbnl. In *2020 IEEE/ACM 2nd Annual Workshop on Extreme-scale Experiment-in-the-Loop Computing (XLOOP)*. IEEE, 1–7.

[9] Ian Foster. 2011. Globus Online: Accelerating and democratizing science through cloud-based services. *IEEE Internet Computing* 15, 3 (2011), 70–73.

[10] Advanced Light Source Photon Science Computing Group. 2025. *splash_flows*. https://github.com/als-computing/splash_flows

[11] Doga Gürsoy, Francesco De Carlo, Xianghui Xiao, and Chris Jacobsen. 2014. TomoPy: a framework for the analysis of synchrotron tomographic data. *Synchrotron Radiation* 21, 5 (2014), 1188–1193.

[12] Pieter Hintjens. 2013. *ZeroMQ: messaging for many applications*. " O'Reilly Media, Inc.".

[13] Zhengchun Liu, Ahsan Ali, Peter Kenesei, Antonino Miceli, Hemant Sharma, Nicholas Schwarz, Dennis Trujillo, Hyunseung Yoo, Ryan Coffee, Naoufal Layad, et al. 2021. Bridging data center AI systems with edge computing for actionable information retrieval. In *2021 3rd Annual Workshop on Extreme-scale Experiment-in-the-Loop Computing (XLOOP)*. IEEE, 15–23.

[14] Shizhao Lu, Tanny Chavez, Wiebke Koepp, Guanhua Hao, Petrus H Zwart, and Alexander Hexemer. 2025. Optimizing inference of segmentation on high-resolution images in MLExchange. *The Journal of Supercomputing* 81, 9 (2025), 1–17.

[15] AA MacDowell, DY Parkinson, A Haboub, E Schaible, JR Nasiatka, CA Yee, JR Jameson, JB Ajo-Franklin, CR Brodersen, and AJ McElrone. 2012. X-ray micro-tomography at the Advanced Light Source. In *Developments in X-ray Tomography VIII*, Vol. 8506. SPIE, 298–311.

[16] James E McClure, Junqi Yin, Ryan T Armstrong, Ketan C Maheshwari, Sean Wilkinson, Lucas Vlcek, Ying Da Wang, Mark A Berrill, and Mark Rivers. 2020. Toward real-time analysis of synchrotron micro-tomography data: accelerating experimental workflows with AI and HPC. In *Driving Scientific and Engineering Discoveries Through the Convergence of HPC, Big Data and AI: 17th Smoky Mountains Computational Sciences and Engineering Conference, SMC 2020, Oak Ridge, TN, USA, August 26-28, 2020, Revised Selected Papers 17*. Springer, 226–239.

[17] Matt McCormick, Paul Elliott, Brianna Major, Sebastien Jourdain (Kitware), Scott Wittenburg, Wei Ouyang, Laryssa Abdala, Don Smyth, Ievgen Popovych, Pablo Hernandez-Cerdan, Benjamin Rombaut, Jade Bilkey, Bertrand Kerautret, and Ido Rosenblum. 2024. *Kitware/itk-vtk-viewer: v14.50.1*. doi:10.5281/zenodo.10680831

[18] Matthew McCormick, Xiaoxiao Liu, Julien Jomier, Charles Marion, and Luis Ibanez. 2014. ITK: enabling reproducible research and open science. *Frontiers in neuroinformatics* 8 (2014), 13.

[19] William L. Miller, Deborah Bard, Amber Boehnlein, Kjiersten Fagnan, Chin Guok, Eric Lançon, Sreeranjani Ramprakash, Mallikarjun Shankar, Nicholas Schwarz, and Benjamin L. Brown. 2023. *Integrated Research Infrastructure Architecture Blueprint Activity (Final Report 2023)*. Technical Report. US Department of Energy (USDOE), Washington, DC (United States). Office of Science; Lawrence Berkeley National Laboratory (LBNL), Berkeley, CA (United States). doi:10.2172/1984466

[20] Carlo Minotti, Alun Ashton, Spencer Bliven, Fredrik Bolmsten, M Novelli, T Richter, Dylan McReynolds, M Leorato, D Van Dijken, and L Schemilt. 2024. Enhancing Data Management With SciCat: A Comprehensive Overview of a Metadata Catalogue for Research Infrastructures. In *19th International Conference on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'23), Cape Town, South Africa, 09-13 October 2023*. JACOW Publishing, Geneva, Switzerland, 1195–1200.

[21] Viktor Nikitin. 2023. TomocuPy–efficient GPU-based tomographic reconstruction with asynchronous data processing. *Synchrotron Radiation* 30, 1 (2023), 179–191.

[22] Viktor Nikitin. 2025. *streamtomocupy*. https://github.com/nikitinvv/streamtomocupy Accessed: 2025-07-30.

[23] Dilworth Y Parkinson, Tanny Chavez, Monika Choudhary, Damon English, Guanhua Hao, Thorsten Hellert, Simon C Leemann, Slavomir Nemsak, Eli Rotenberg, Andrea L Taylor, et al. 2024. AI@ ALS Workshop Report: Machine Learning Needs at the Advanced Light Source.

[24] Hannah Parraga, John Hammonds, Steven Henke, Siniša Veseli, William Allcock, Benoit Côté, Ryan Chard, Suresh Narayanan, and Nicholas Schwarz. 2023. Empowering scientific discovery through computing at the advanced photon source. In *Proceedings of the SC'23 Workshops of the International Conference on High Performance Computing, Network, Storage, and Analysis*. 2126–2132.

[25] Will Schroeder, Ken Martin, and Bill Lorensen. 2006. The visualization toolkit, 4th edn. Kitware. *New York* (2006).

[26] Advanced Light Source. 2025. ALS-U Overview. https://als.lbl.gov/als-u/overview/. Accessed: July 31, 2025.

[27] Laurie Stephey, Shane Canon, Aditi Gaur, Daniel Fulton, and Andrew J. Younge. 2022. Scaling Podman on Perlmutter: Embracing a community-supported container ecosystem. In *2022 IEEE/ACM 4th International Workshop on Containers and New Orchestration Paradigms for Isolated Environments in HPC (CANOPIE-HPC)*. 25–35. doi:10.1109/CANOPIE-HPC56864.2022.00008

[28] Jana Thayer, Daniel Damiani, Mikhail Dubrovin, Christopher Ford, Wilko Kroeger, Christopher Paul O'Grady, Amedeo Perazzo, Murali Shankar, Matt Weaver, Clemens Weninger, et al. 2019. Data processing at the linac coherent light source. In *2019 IEEE/ACM 1st Annual Workshop on Large-scale Experiment-in-the-Loop Computing (XLOOP)*. IEEE, 32–37.

[29] U.S. Department of Energy. 2023. U.S. Department of Energy Public Access Plan 'Ensuring Free, Immediate and Equitable Access' to the Results of Department of Energy Scientific Research. https://doi.org/10.11578/2023DOEPublicAccessPlan

[30] Siniša Veseli, John Hammonds, Steven Henke, Hannah Parraga, and Nicholas Schwarz. 2023. Streaming data from experimental facilities to supercomputers for real-time data processing. In *Proceedings of the SC'23 Workshops of the International Conference on High Performance Computing, Network, Storage, and Analysis*. 2110–2117.

[31] Marco Voltolini. 2021. In-situ 4D visualization and analysis of temperature-driven creep in an oil shale propped fracture. *Journal of Petroleum Science and Engineering* 200 (2021), 108375.

[32] Marco Voltolini and Jonathan Ajo-Franklin. 2020. Evolution of propped fractures in shales: The microscale controlling factors as revealed by in situ X-Ray micro-tomography. *Journal of Petroleum Science and Engineering* 188 (2020), 106861.

[33] Chunpeng Wang, Ullrich Steiner, and Alessandro Sepe. 2018. Synchrotron big data science. *Small* 14, 46 (2018), 1802291.

[34] Samuel S Welborn, Chris Harris, Stephanie M Ribet, Georgios Varnavides, Colin Ophus, Bjoern Enders, and Peter Ercius. 2025. Streaming Large-Scale Microscopy Data to a Supercomputing Facility. *Microscopy and Microanalysis* 31, 1 (2025), ozae109.

[35] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. 2016. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific data* 3, 1 (2016), 1–9.

[36] Kevin G Yager, Pawel W Majewski, Marcus M Noack, and Masafumi Fukuto. 2023. Autonomous x-ray scattering. *Nanotechnology* 34, 32 (2023), 322001.

[37] Naweiluo Zhou, Giorgio Scorzelli, Jakob Luettgau, Rahul R Kancharla, Joshua J Kane, Robert Wheeler, Brendan P Croom, Pania Newell, Valerio Pascucci, and Michela Taufer. 2023. Orchestration of materials science workflows for heterogeneous resources at large scale. *The International Journal of High Performance Computing Applications* 37, 3-4 (2023), 260–271.