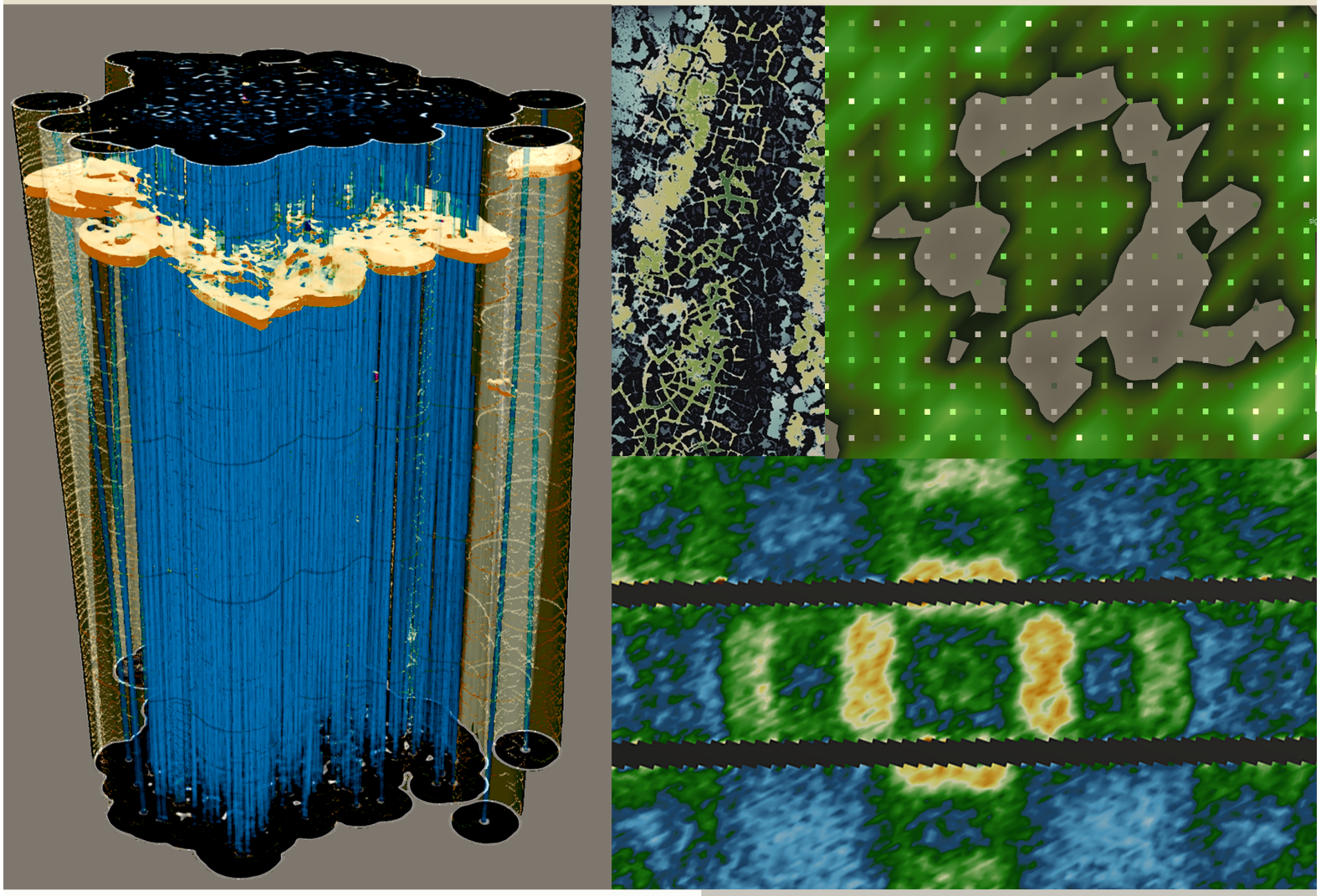
Report of the
DOE Workshop on

Management, Analysis, and Visualization of Experimental and Observational Data

The Convergence of Data and Computing



U.S. DEPARTMENT OF ENERGY | Office of Science

September 29th - October 1, 2015
Bethesda, MD

Management, Visualization, and Analysis of Experimental and Observational Data (EOD)
The Convergence of Data and Computing
Workshop Final Report

Office of Advanced Scientific Computing Research, DOE Office of Science
Bethesda, Maryland
September 29–October 1, 2015

This work was supported by the Director, Office of Science, Office and Advanced Scientific Computing Research, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

This is LBNL report LBNL-1005155.

Contents

Workshop Overview and Findings

Executive Summary

The Department of Energy (DOE) Office of Science (SC) operates dozens of national science user facilities that span many di erent disciplines. These facilities include accelerators, colliders, supercomputers, light sources and neutron sources, as well as facilities for studying the nanoworld, the environment, and the atmosphere. In Fiscal Year 2014 over 33,000 researchers from academia, industry, and government laboratories, spanning all fifty states and the District of Columbia, utilized these unique facilities to perform new scientific research. Each of these facilities generates vast amounts of scientific data, and the rate, size, and complexity of this data is rapidly increasing, thanks to advances in technology. A growing concern, which motivates this workshop, is the likely significant adverse impact on science programs that will result without significant advances in the capabilities needed to manage and gain knowledge from these collections of data. The purpose of this workshop, held 29 September 2015 through 1 October 2015 in Bethesda, MD, is to help the Advanced Scientific Computing Research (ASCR) and research community better understand needs related to the management, analysis, and visualization of experimental and observational data (EOD) collected and generated by experimental and observational science projects (EOS) at Office of Science user facilities.

The science needs articulated in this report, along with the findings, recommendations, and detailed discussion of issues, collectively are consistent with and show opportunity for cultivating a research, development, and deployment path that takes steps towards realizing the vision articulated in the National Strategic Computing Initiative (NCSI) [1, 2] and the Big Data Research and Development Initiative [3, 4]. Specifically, the science use cases reveal a trend towards the *convergence of data and computing*: data- and compute-centric needs and opportunities are increasingly intertwined, interrelated, and symbiotic. Advances in our ability to collect data in turn require advances in computational capabilities to understand, preserve, share and make optimal use of data, and can even favorably impact the quality and value of science we perform by improving the quality of data we collect.

This workshop report consists of input from a set of representatives from DOE EOS facilities and researchers in mathematics and computer science. The findings, drawn from use cases that articulate science drivers, indicate acute and urgent needs. This report articulates a path forward for meeting those needs, a path that includes advances in mathematics, computer and data science, as well as advances in and the use of HPC computational and networking infrastructures. One major theme recurring in the use cases is that individual EOS projects are presently pursuing their own independent paths towards meeting data-centric challenges, which results in the duplication of e ort and increased costs across the entire program. EOS projects, and the EO community as a whole, would benefit immensely from a coordinated, program-wide e ort that targets fosters research, development, deployment, and sustainment of data-centric software tools and infrastructure for meeting needs in computing, data storage, curation, archival, and dissemination.

Findings and Recommendations

Findings

1. **All EOS projects represented at this workshop struggle to keep up with the demands and opportunities that a flood of data o ers.** Data acquisition rates for individual EOS projects are rapidly approaching tens of petabytes per year (see §23), which sums to multiple exabytes per year across all EOS projects. This condition of rapidly growing data size, complexity, and diversity challenges and impacts all EOS projects represented at this workshop. The complexity of the data, new challenges in analytics and visualization, difficulties in capturing sufficient metadata, and ease-of-use problems are impediments to the use and adoption of many types of data-centric tools and infrastructure, hampering the e ort to harness the wealth of data in the service of scientific discovery.
2. **Meeting the challenges of the explosion of data from EOS projects requires computational platforms, networking, and storage of greater capacity and lower latency, along with software infrastructure suited to their needs.** However, existing HPC platforms and their software tools are designed and provisioned for high-concurrency HPC workloads, single-project data products, and comparatively simpler data needs. The result is a significant gap between the needs of EOS projects and the current state of the art in computational and software capabilities and resources.
3. **EOS projects increasingly rely on low-latency, fast-turnaround resource response to meet data-centric needs.** With time-sensitive responsiveness, experimental design and operation becomes more efficient as EO researchers are able to refine and guide experiment parameters to converge on better science results. Facility support for such timely, human-in-the-loop capabilities is currently *ad hoc* at best.
4. **Scientific data is increasingly at risk of being unusable.** Without adequate metadata, scientific data has limited usefulness because its origins are undocumented and unknown, thereby limiting the ability to validate results or to make use of such data for other purposes. However, today the capture of this critical information often relies on manual, non-digital and non-sharable approaches, hindering scientific discovery particularly in increasingly high-velocity, high-volume data environments.
5. **Collaboration and sharing of data, tools, and methodologies are central to modern EOS projects, yet there is insufficient infrastructure to facilitate such interactions.** The obstacle is not simply data transfer, but rather a lack of widely used tools to produce and consume well-characterized data collections that include the desired level of annotation, metadata and provenance. The process of collaboration requires an ability to share software tools, source code, data models and formats and workflows that are reproducible. Beyond established collaborations, there is a clear need to share tools and approaches between groups and disciplines to minimize the unnecessary duplication of e ort. In most cases, existing tools are inadequate or too difficult to use.
6. **EOS projects are impeded due to significant "data lifecycle" needs that are largely unmet.** While some stages in the data lifecycle are well supported, others are not. Data collected by observation or experiment, along with the software tools used for its analysis, have a potentially long lifespan and a potentially large set of consumers, but presently there are no solutions nor approaches within the DOE SC that are generally and broadly applicable for data curation, quality management, and

long-term distribution or dissemination. At the same time, data retention policies at SC computing facilities are not designed for long-term retention nor for widespread dissemination.

7. **EOS projects can benefit greatly from coordinated e orts to design, implement, deploy, and sustain critical software tools for working with data that target EOS data needs and workflow patterns.** Software is a critical element for all EOS projects in all aspects of working with data and in meeting the challenges of increasing data size and technology complexity: it is used for collecting data, processing and analyzing data, for preparing data products, for automating complex multistage operations that may span distributed resources. A recurring theme present in the use cases is that software design and development is most often an activity conducted within a particular EOS project or facility as each project focuses on meeting its own particular needs as quickly as possible. The results include increased overall cost for software development from redundancy of e ort, software that exists "in isolation" from other EOS projects due to the absence of established practices for curation and dissemination, and software designs and implementations that may not be highly usable or sufficiently flexible to be adaptable to other EOS projects or emerging computational technologies. Software is "digital data" that needs to undergo the rigors of curation, in the same way as data from experimental and observational sciences, to facilitate long-term archival preservation and widespread dissemination.
8. **The highly specialized nature of skills and expertise in the data sciences and their application to EOS problems raises concerns about workforce training, development and retention.** This concern is made more acute by the growing competition for data science specialists in all areas of commerce and industry.

Recommendations

1. **Address the challenges posed by the growing size, increasing data rate, and complexity of data through concerted, dedicated and shared e orts.**
   (a) Using a multidisciplinary approach, carry out research into new methods for mathematics, analytics, visualization, collaboration, and data management that targets key data challenges in EOS; and coordinate these research activities with broader software tools and infrastructure to facilitate their deployment, sustainment, and use by EOS projects and facilities.
   (b) Adopt and integrate modern data storage and access technologies that both accommodate present and future data size and rate values, and that are suitable for use in key EOS data-centric processing workflows.
   (c) Cultivate multidisciplinary teams and programs that focus on software solutions to data-centric challenges that are broadly applicable beyond a single EOS project.
2. **Evolve HPC computational facilities to include focus on the needs of the EOS community.**
   (a) Identify and prioritize EOS-centric operational and resource needs for major HPC computational facilities and networking infrastructure. Reconsider facility metrics and priorities in response to the requirements of the EOS community. At the same time, study the hardware and software architectural implications of EOS data needs.
   (b) At HPC computational facilities, consider approaches for providing resources, along with suitable operational policies for their use, that are attuned to the needs of the EOS community.
   (c) Evaluate alternatives for providing to the EOS community long-term data storage and archival services that support advanced search and subset capabilities as well as mechanisms that enable stratified and selective public distribution of data.

(d) Evaluate strategies for enabling EOS projects to take advantage of major HPC facilities to service data-centric workloads, including but not limited to computation with real-time or interactive response, data storage, archival, and dissemination.

3. **Develop solutions to meet EOS needs for fast-response, low-latency, high-throughput (time-critical), data-centric workloads.**

   (a) Develop a systematic, end-to-end understanding of time-critical EOS needs that includes the appropriate metrics and that takes into account human-in-the-loop scenarios.

   (b) With an eye towards addressing the needs time-critical data-centric workloads, assess the applicability of factors like HPC architectures, HPC software infrastructure, programming models and runtime systems.

   (c) Develop a set of facility requirements, siting strategies, and appropriate use and operations policies aimed at meeting the needs of time-critical EOS workloads.

   (d) In support of time-critical use scenarios, cultivate and deploy new methods and practices, such as those that reduce data size or accelerate key computational stages in the processing pipeline.

4. **Improve the EOS productivity with new resilient solutions to automate data-intensive processing pipelines.**

   (a) Develop a deeper understanding of the workflow usage commonalities and execution patterns in the EOS ecosystem that considers data generation, movement, processing, sharing, and dissemination.

   (b) Develop solutions for optimizing quantitative metrics, such as performance, reliability, scalability and throughput, while addressing qualitative metrics, such as as learnability, usability, manageability and transparency.

   (c) Take steps to promote reusability and reproducibility of workflows and associated methods across EOS projects and HPC computational facilities.

5. **Develop a better understanding of EOS metadata needs and solutions, and develop and deploy software tools for meeting metadata needs of the EOS community.**

   (a) Develop a systematic understanding of how data are to be used, present and future, across EOS projects, and focus metadata-facing R&D e orts accordingly.

   (b) Conduct research and development on the management and use of scientific metadata to enable cross-community sharing, semantic understanding, and advanced methods for scientific search.

   (c) Strive to achieve better integration and automation of metadata capture, and event and feature tagging, into scientific workflows.

   (d) Develop tool sets for capturing, storing, and managing metadata that can be widely deployed.

6. **Expand capabilities for the collaboration and sharing of EOD and tools.**

   (a) Develop a systematic understanding of EOS needs as they pertain to collaborations of science communities, particularly data sharing needs and practices.

   (b) Develop approaches that simplify and facilitate collaboration in general, and share data, software, and workflows in particular.

   (c) Develop and deploy tools that are data- and metadata-driven to enable use without highly specialized expertise.

7. **Identify and fill gaps in data lifecycle, reproducibility, and curation support.**

   (a) Develop an understanding of broad needs and requirements, an assessment of technologies available for meeting those requirements, and the extent to which this set of needs and requirements are amenable to common solutions within and across institutions and disciplines.

   (b) Craft and implement an R&D, deployment, and sustainability road map for tools and operational procedures that are broadly applicable across EOS projects and user facilities that target EOS needs in data lifecycle, reproducibility, and curation.

   (c) Develop strategies for provisioning infrastructure for the long-term maintenance, preservation, and distribution of curated data and software.

   (d) Assess the possibility of provisioning centralized DOE-SC-wide facilities for data archival and retrieval, including mechanisms for usage-based cost recovery.

8. **Expand e orts aimed at understanding and filling gaps in the software ecosystem of EOS projects and user facilities.**

   (a) Develop a systematic understanding of the broad and diverse data-centric software needs of EOS projects.

   (b) Cultivate software R&D projects focusing on EOS needs that follow best practices in software engineering that include mechanisms for long-term software archival and curation, dissemination, support, and user training.

   (c) Emphasize design, development, and deployment, and sustainment of software that is highly usable, rapidly customizable, reusable, straightforward to deploy across scales ranging from major HPC facilities to desktop, laptop, or portable platforms, and that adopts best practices in software development, engineering, and maintenance.

   (d) Identify opportunities for a broader coordination of data-centric software for EOS that target key needs, such as software reuse and sharing across EOS user facilities and projects, and software curation and dissemination.

   (e) Identify and establish practices for software archival, curation, dissemination, and long-term support.

9. **Develop and nurture a data science workforce.**

   (a) Prioritize the role of data science activities in multidisciplinary teams that endeavor to develop and deploy methods that target the data needs of the EOS community.

   (b) Recognize and reward the special skills and roles of this group within the research community, EOS projects, and funding agencies.

   (c) Provide career paths that reward and recognize research, development, deployment accomplishments that have positive impact on EOS projects.

Amplification of Findings and Science Mission Drivers

The Challenges of Exploding Data Size, Rate, Complexity and Diversity

Data size and rate of collection at science user facilities is growing at a rapid rate. Each of the EOS use cases in this report provides details about expected and anticipated growth in data rates. These individual tables are repeated in consolidated form in Appendix 23. Integrating across all these summaries of projected growth rates, we see a future where individual facilities, of which there are dozens, are each generating collections of data in the range of 1–50 PB per year. These projections suggest, when integrating across the entire program, that these *science user facilities will be soon collectively acquiring exabytes of data per year.* In the present, these projects are having difficulty coping with the data they collect, and help is urgently needed now to be prepared for the future.

All EOS projects represented at this workshop are having difficulty in keeping up with the demands and opportunities that the flood of data o ers. The complexity of the data, new challenges in analytics and visualization, difficulties in capturing sufficient metadata and ease-of-use problems are impediments to use and adoption of many types of data-centric tools and infrastructure, hampering the e ort to harness the wealth of data in the service of scientific discovery.

> A key limitation today is our [in]ability to analyze and visualize the acquired data due to its volume, velocity and variety (§18.2.1).

One of the primary drivers for increasing data size is the increase in the resolution of the instrument sensors. Data rate increases at existing beamlines will come from new and improved detectors, as well as from increases in flux and brightness due to upgrades in the storage ring, beamline optics, and end stations (§14.1.2, §18.2.2). For some projects, the growth rate is urgent; at the Advanced Photon Source (APS), they expect a growth on the order of one order of magnitude in the coming months, followed by another two to three orders of magnitude increase that will result from the APS upgrade, which will permit multiple techniques to be applied simultaneously to a single sample (§18.2.2).

The data size and velocity problem is compounded when a given EOS community relies on multiple instruments, such as the Cosmic Frontier projects, which carry out sky surveys using multiple instruments. These are expected to produce data estimated to be on the order of hundreds of petabytes in size. These data will be made available to a community of researchers over a long period of time (§21.1). Survey data is very valuable with a very long shelf life and is mined and analyzed in a number of ways, depending on the science use case.

These data generation volumes extend beyond issues in processing and storage, but also in data transfer, particularly in experiments that rely on real-time feedback to the tool operator. This problem is complicated even further by the fact that many of the experiments summarized may happen concurrently with parallel data flows coming from independent detectors (§17.1.2).

Another challenge associated with increased data size and complexity is the need to better support data integration and data discovery through the collection and management of metadata and derived data products associated with experimental and simulation data (§19.1.5).

An ongoing concern in EOS projects is that the data they collect is free from error, and that it focuses on the specific science objective. Here, we see a clear convergence between computing and data, where computational methods can be brought to bear to ensure the best possible data are collected during an experiment. In some cases, errors occur during data acquisition. These errors can be mitigated/corrected after taking data through advanced algorithms that can model the dynamical e ects of the acquisition instrument to produce a data set with minimized error. (§18.2.1)

A related concern is the loss of science and opportunities for science discovery due to data loss. One example is the Cosmic Frontier projects where data loss can occur in studies of transients because of possible inefficiencies in detection technology, classification algorithms, and lack of follow-up resources. Other issues that prevent making use of the complete data set are technical issues such as lack of understanding of foregrounds, modeling the atmosphere, detector noise, etc. (§21.2.3).

While coping with the increasing size and rate of data inflows from experiments and observations is a challenge, there is a corresponding set of challenges at the other end of the data pipeline. EOS projects typically also produce "data products" that are derived from raw experimental or observational data, and in many cases, from the results of numerical calculations. Some data products are produced for individual users (§11.1.1, §14, §18), while other data products are intended to be used by entire communities (§21.2, §12.1) or as reference data sets (§20.1.1). For data products, having a clear record of information about the data (metadata) is essential in order for these data to be useful and usable, and there is clear need for a long-term plan for addressing the archival, curation, and dissemination of data products.

EOS Projects' Use of Large-Scale High Performance Computing Facilities

EOS projects' use of tools and facilities, which are designed for HPC workloads, have realized varying degrees of success. Meeting the challenges of the explosion of data from EOS projects requires computational platforms, networking, and storage of greater capacity and lower latency, along with software infrastructure suited to their needs. However, existing HPC platforms and software tools are designed and provisioned for high-concurrency HPC workloads, single-project data products, and comparatively simpler data needs.

EOS projects look to large-scale HPC computing facilities to help serve workloads that can be characterized as: requiring fast turnaround for computing tasks, having processing pipelines that are distributed in nature and involve the movement of a significant amount of data, long-term storage of data and providing access to data to a potentially diverse set of stakeholders and consumers.

The issue of fast turnaround is so significant that it receives its own section in these findings. In brief, the issue is that EOS projects like beamlines require computational resources within minutes or perhaps seconds when data are available and cannot abide with the queued structure employed on leadership machines (§18.2.2).

The EOS use cases in this report describe variants of data handling and processing activities that can be characterized as distributed computing models. The typical design pattern involves first collecting data at

the instrument, performing some processing close to the instrument, moving data to a large-scale facility for more lengthy calculations and preparation of data products, then dissemination of data products. The way each project implements this pattern varies according to their needs and available resources.

For example, the Deep Underground Neutrino Experiment (DUNE) experiment presently uses a combination of local, on-site computing and HPC facilities at Fermi National Accelerator Laboratory (FNAL), which also is expected to host a full replica of data recorded by the prototype instrument. DUNE plans to keep full data replicas elsewhere for redundancy, as well as to opportunistically leverage computing resources, including those outside of DOE. DUNE is targeting the design and development of project-wide software infrastructure that aims to maintain portable and accessible software that can be used at any particular institution and run transparently on modern Grid and/or cloud resources as part of a distributed processing data-centric workflow (§19.1.2).

> Procedures for moving data from place to place, including tools for automating resilient workflow for orchestrating distributed data-related operations are a bottleneck (§12.2).

There is a clear need for community- or facility-centric data repositories for data archival, sharing; with substantial bandwidth to the stored data, and easy interface for interacting with the data analytics. This needs to be massively parallel, a combination of visualization and various analysis tools (§20.1.3). It is very likely that DOE facilities (both ASCR and HEP) will take on a significantly larger role in data archiving, transfer, and analysis. It is also possible that commercial cloud resources will become a major resource in these areas—although several outstanding questions remain (e.g., cost models, data archiving and transfer); this disruptive possibility needs to be continuously explored. The main new hardware trend of interest for DOE facilities—in the relatively near-term—is the evolution and integration of HPC systems within a data-centric usage model (§21.3).

The needs of a data sharing site are quite distinct from one designed to store or analyze data. Data sharing software must have robust features in searching for specific data types and for evaluating their relationships to people, studies, scientific fields and published results (§11.2).

All of these factors are somewhat at odds with how platforms and software infrastructure are architected, as well as with operational policy: to service long-running, high-concurrency jobs. We examine these issues in more detail from an HPC facilities perspective (§6).

Time-critical Data Needs

Many projects have time-critical data needs. These projects require a low-latency, high-throughput response from infrastructure for data movement, analysis and processing and storage. However, computing platforms available to these researchers are insufficient in capacity or turnaround. The lack of large and capable facilities tuned to EOS needs are common across many disciplines.

Many EOS projects use, or hope to use, large-scale HPC platforms and high-speed networking to do real-time processing of experiment data. The high-throughput, fast turnaround enables on-the-fly adjustment of experiment parameters while the experiment is in progress, thereby creating the possibility of maximizing scientific results (§14.1.3, §16.1.2, §17.1.2, §18.2.1, §19.1.2, §19.2).

At the very first stages of the analysis workflow, scanning electron microscopy (SEM) projects are interested in collecting full detector response at the fastest meaningful rates in order to assess tool performance and adjust parameters on-the-fly. Additionally fast visualization schemes would be of use to monitor the sample and quality of the output signal (§17.1.2).

> Predicting the optimal scanning parameters, such as detector exposure time, number and optimal angular position of the projections could optimize data collection schemes and ultimately provide better quality data. . . . Besides predicting the optimal scanning parameters, the analysis of the resulting data then becomes the next bottleneck preventing near-real-time error detection or experiment steering (§18.2.1).

A recurring theme in these EOS projects is the potential for increasing the quality of science by being able to perform key data-intensive computations quickly so as to adjust experimental parameters on-the-fly. In some cases, these computations can be performed on platforms close to the experiment. In other cases, the computational power required exceeds that available locally, and these projects look to resources at HPC facilities. In turn, such a distributed, data-intensive workflow will also place demands on networking infrastructure for the fast movement of large volumes of data.

The Risk of Unusable Data

Scientific data is increasingly at risk of being unusable, and, hence, at risk of being lost forever. Without adequate metadata, scientific data has limited usefulness because its origins are undocumented and unknown, thereby limiting the ability to validate results or to make use of such data for other purposes. However, today the capture of these critical information often relies on manual, non-digital and non-sharable approaches, hindering scientific discovery particularly in increasingly high-velocity, high-volume data environments.

In some projects, data-centric operations—management, analysis, movement, distribution—are the responsibility of an individual user, with whatever limited knowledge and capability is available to them. As a result, only a fraction of collected data is every analyzed, and only a fraction of that data is ever published and made available for community-wide use (§17.1.1).

> One very real problem is that presently the data is almost never usable by anyone other than the original group that generated it. This problem must be solved if making data publicly available is intended to have any useful purpose (§11.1.1).

One very real problem is that, at present, data is difficult to use by anyone other than the original group that generated it. This problem must be solved if making data publicly available is intended to have any useful purpose. In addition, much necessary metadata is never collected because of the lack of understanding of what is required for data sharing by the primary investigator and the lack of easy-to-use tools to capture it. The overall cost and complexity of metadata recording and consolidation is currently prohibitive, which is the primary reason it is rarely collected. Unfortunately, this means that the associated data cannot be easily discovered or reused (§11.1.1). Systematic collection of the metadata that describes the provenance of stored data is typically inadequate, limiting the integrity, traceability and reproducibility of research products.

> . . . relevant data should be made available to the scientific community after some amount of time. But more than data preservation is required—proactive data curation is necessary for the data to be really useful. . . . The benefit of curation would be to reduce duplication of e ort in data creation, but also for the re-use of data for further high quality research. (§14.2)

There is interest in having access to data after the current research is published. Such access needs to ensure that enough metadata is stored so that the data can be analyzed appropriately. There is a need to capture the reason why certain aspects of an analysis or data transformation or reduction operation was performed (§16.1.2). This information, the metadata, needs to be archived with the data so that subsequent access is useful, and can be utilized by researchers beyond the group that acquired it originally (§16.2).

Collaboration and Sharing are Activities Central to EOS Projects

Collaboration and sharing of data, tools, and methodologies are central to modern EOS projects, yet there is insufficient infrastructure to facilitate such interactions. However, common tools and methodologies for sharing and collaboration in data-intensive sciences have not been widely developed, deployed, or adopted. The limit is generally not simply data transfer, but rather a lack of widely-used tools to produce and consume well-characterized data collections that include the desired level of annotation, metadata and provenance. Collaborations also require an ability to share software tools, source code, data models and formats and workflows that are reproducible. Beyond established collaborations, there is a clear need to share tools and approaches between groups and disciplines to minimize the unnecessary duplication of e ort. In most cases, existing tools are inadequate or too difficult to use.

By their nature, the mission focus for EOS projects is to collect data, and to share it. This theme is present in all the use cases present in this report. The projects di er in some key ways: some projects' immediate focus is on sharing data with a primary principal investigator (PI) or PI group (e.g., §11, §17, §14), while others focus on sharing data with larger communities (e.g., §21, §13). While making data accessible for download over the Internet lowers the barrier to accessibility for a potentially large number of consumers, doing so is only a small part of a larger landscape of collaboration and sharing.

> Understanding the process of how science is actually done, what information needs to be captured and where the data is generated are key issues that must be addressed to enable e ective data sharing (§11.2).

One concept that is central to achieving the ability to share data and tools is the idea of community-centric, or "standard" data models and formats for both data and metadata. The climate community, for example, has realized a degree of success in sharing data as well as software tools for working with data, due to its use of a data model or format that has broad community support (§12.1, §13.1.3). This idea is identified as a need or an impediment in several use cases (§21.4).

The use cases provide several compelling reasons why collaboration and sharing is important. First, sharing software has the potential e ect of reducing costs, particularly of software development (§18.3). The idea is that redundancy of e ort—software development—is reduced when key methods and tools can be reused across di erent projects. Sharing data, particularly curated data, would be to reduce the duplication of e ort in data creation, as well as for data re-use for further high-quality research (§14.2). Another benefit would be that it could lead to more algorithms and software being made available to the community, as researchers write code that can be benchmarked and used against curated data.

> Current technologies are inadequate for sharing [...] data between group members. The community needs a more fluid means for sharing data and working together (§11.2).

The use cases identify several di erent ideas that are needs for or impediments to collaboration and shar-

ing. One is that the issue of data and software sharing does not have program-wide visibility. As a result, progress in this space is *ad hoc*, with solutions for distributing data (§11.1.3) or software (§18.2.2) emerging on a per-facility or per-PI basis, with little or no coordination. The result is that there exist many di erent sources of data and software (duplication of e ort) and there is a high barrier to finding data or software. Impediments that are most detrimental are related to the issues of data sharing and collaborating in large groups, methodological transparency, and dissemination and archival capabilities (§11.2). Data and/or software that is "custom" and not curated is unlikely to be widely used (§17.1.1). Better methods—interfaces and software tools, infrastructure—are needed to search and subset data without having to download an entire data set (§13.1.4).

> Impediments that are most detrimental are related to the issues of data sharing and collaborating in large groups, methodological transparency, and dissemination and archival capabilities (§11.2).

There is a deep interplay between the topic of collaboration, which is the subject of Section 1, and the related but orthogonal topics of the overall data lifecycle, the usability of data and the associated challenges of metadata/provenance capture and long-term data archival and curation, and EOS's use of computing and data facilities. The interactions between these di erent focus areas is made more challenging by the rapid rate of growth in data size and the rate of data acquisition. Stated di erently, successes in these related areas are building blocks for success in the area of collaboration and sharing.

The Data Lifecycle Needs in Environmental and Observational Science are Not Being Met

EOS projects have "data lifecycle" needs that are significant, well defined, and that go well beyond what is provided by the current set of programs and projects in the ASCR computing facilities and research portfolio. EOD can have a long lifespan, yet there is no program-wide view or approach for the long-term curation, storage, and dissemination of such data; one EOS project indicated that it relies on whatever capabilities are provided by journals in association with publications as its solution to this problem.

The term data lifecycle refers to all stages of data collection, movement, processing, analysis, management, curation, and sharing. Data collected by observation or experiment has a potentially long lifespan, and a potentially large set of consumers, but there presently is no solution or approach for data curation, quality management, and long-term distribution within DOE SC that is generally and broadly applicable. At the same time, data retention policies at SC computing facilities are not designed for long-term retention nor for widespread dissemination.

> . . . our only archival process right now is that provided by the published journal (§16.1.1).

Two key motivations for retaining data sets for a long period of time are for having a reference data set for use in evaluating the e ectiveness of new methods over time, and for the opportunity for new discoveries not originally foreseen at the time the data was collected. Over the years, some data sets produced by simulation will emerge as a community reference. For such collections, which will be used by many di erent authors in refereed publications, reproducibility of these analyses will become another reason to keep the data, even when better and higher resolution alternatives become available (§20.1.3). In tomography, the resulting tomogram is of comparable size to the raw images, which are also usually retained for the pur-

poses of comparing the results of different tomographic reconstruction algorithms (§18.2.2). Results from projects like sky surveys may initially be focused on a few key science missions, but over time, a diverse set of science activities can be carried out with substantial discovery potential (§21.2).

Current strategies for managing (accessing, processing, and keeping track of) the large number of data products are awkward at best, requiring a combination of methods (§12.2). User facilities like the APS do not provide a centralized and robust long-term data archive, since this service is categorized as a user responsibility (§18.2.4). Most user facilities have no explicit method for long-term archival and curation, and this is identified as an impediment (§21.4). In the future, science user facilities may be called upon to provide long-term storage and archival services (§18.2.4). One stop-gap approach for long-term archival is to rely on that provided by the journal where a given paper is published (§16.1.1). A welcome addition in the data universe would be a centralized DOE facility that provides a mechanism for data archival and retrieval, that could be provided as an option to users at cost (§18.3).

> Providing more access to the data, in a manner that can be used by more scientists, will improve efficiency, increase the impact of the science, and result in more papers per experiment (§16.1.2).

The issues related to data lifecycle management are broad, and cut across many different areas. We have identified challenges and research needed in areas germane to this topic: the automation of processing stages and automated data movement in EOS (§7), data storage and retrieval (§8), metadata and provenance (§9), software engineering and infrastructure (§3), data curation (§10), collaboration (§1), and interaction with computing service facilities (§6).

The Central Role of Software in EOS Projects

Software is a critical element for all EOS projects in all aspects of working with data and in meeting the challenges of increasing data size and technology complexity. It is used for collecting data, processing and analyzing data, for preparing data products, and for automating complex multi-stage operations that may span distributed resources.

An important outcome of this workshop is the recognition of common needs across all the science domains. While the computing needs of EOS projects vary from one project to the next, it is the case that all EOS projects need computing, data storage/dissemination, along with a sustainable software ecosystem that can evolve over time to accommodate its data-centric requirements. This finding suggests that priority attention should be directed towards approaches that develop and support solutions that can be widely used by many EOS projects and facilities.

The EOS use cases in this report describe variants of data handling and processing activities that can be characterized as distributed computing models. The typical design pattern involves first collecting data at the instrument, performing some processing close to the instrument, moving data to a large-scale facility for more lengthy calculations and preparation of data products, then dissemination of data products. The way each project implements this pattern varies according to their needs and available resources. This theme is present in use cases from projects that collect or produce data from instruments (§11, §17, §21), sensors (§13), light sources (§14, §15, §18), neutron sources (§16), detectors (§19), and computations (§12, §20).

> Each beamline operates with unique capabilities and an independent scientific mission. ...Computational needs and strategies may di er considerably across beamlines, but computation is required for nearly every aspect of the facility (§18).

Software methods, such as advanced algorithms for analysis, play a key role in improving the quality of data collected during an experiment, thereby improving the efficiency and quality of science. The issue of fast turnaround is so significant that it receives its own section in these findings. In brief, the issue is that EOS projects like beamlines require computational resources within minutes or perhaps seconds when data are available and cannot abide with the queued structure employed on leadership machines (§18.2.2).

Because of the central role that software plays in nearly all aspects of working with data, EOS projects are particularly vulnerable to inefficiencies and increased costs that can result from software-related issues. For example, inefficiencies in time result when data-centric pipelines and data movement activities must be executed manually rather than being automated and resilient (§12.2); inefficiencies in cost can result when a customized software component is created for one user but is not readily customizable or applicable to other users in the same facility (§14.2, §18.2.2), or across other science facilities.

> The biggest challenge to the facility is how to create the scientific software needed to run it: software for improving the experimental process; for implementing beamline data movement and reduction workflows; to perform preliminary quality assurance, visualization and reduction; for data analysis and interpretation; for automating analysis workflows and distribution to users (§18.2.2).
>
> The most serious impediment the APS encounters is a lack of a DOE-wide view of software needs across the BES mission. Since each lab has its own portfolio of responsibilities, it devotes resources to those goals (§18.3).

Software technology also plays a key role in encapsulating complexity and as an enabling technology. EOS projects want and need to be able to make use of advances in computational architectures, such as using HPC platforms for performing data-centric operations on larger data and with a faster turnaround. However, developing software for those platforms is often beyond the reach of a typical scientist-developer who may not have HPC software development skills. When it comes to the development of HPC code, there are fewer tools that ease the process for scientist-software developers (as opposed to computational experts) to transition from prototype code to HPC production code (§18.3). The same idea extends to other areas of technology, such as creating data-centric pipelines that span distributed resources.

Increasingly, both simulations and experimental data analysis are elements of integrated workflows, which should resiliently automate key components of the data-handling pipeline, from collection to processing, analysis, archival, and dissemination. Many contemporary EOS projects articulate the need to combine computing with the experiment in real time, so as adjust experimental parameters on-the-fly to obtain the best possible data and science result from the experiment. Meeting these challenges will require more powerful computing and networking infrastructure combined with a capable, robust and sustainable software ecosystem focusing on EOS needs. The flood of data available now and in the near future presents an opportunity that can be met only through concerted, coordinated, and sustained e orts to improve the software tools, methods, and facilities (computing, data) available to the EOS community.

Software is "digital data" that needs to undergo the rigors of curation, in the same way as data from experimental and observational sciences, to facilitate long-term archival preservation and widespread dissemination. Like other forms of digital data, to be useful, software needs to have associated metadata along with documentation and examples of use. To be long-lived, it needs to be supported, maintained, and

disseminated, something that is often not a part of the cost model. Several use cases pointed to the desire to distribute software with data, to facilitate in the usability of data and to promote the repeatability of results as well as promote the use of reference data and methods (§16.1.1, §14.2, §16.1.2, §20.1.3). One use case pointed out their only avenue for doing so is to rely on the archival capabilities provided by the journal where results are published (§16.1.1). The issues and motivations related to software curation, preservation, and dissemination are similar to those for other types of scientific data.

Workforce Development and Retention

Arguably, the single most precious resource we have in the sciences is our personnel. As such, the notion of workforce development is an ongoing process for not only present sta , but also the sta of the future.

A recurring theme in the science use cases is the value of multidisciplinary groups of researchers working together to solve data-centric challenges. In such teams, it is often useful for a computer scientist to have some background and knowledge of specific science applications, and vice versa. Generally, those having such a dual background is the exception, and so some amount of professional training and development (boot camps, intensive courses) is required to help fill these kinds of gaps (§17.1.5).

Workforce retention issues are multi-faceted and can be challenging to address. At least one use case calls out that there is a lack of an adequately trained workforce combined with inadequate or insufficient career paths for computationally-oriented scientists (§21.4).

Related to career paths is the notion of compensation, rewards, and valuing the contributions of those working on data-centric challenges. Given that we are increasingly a data-driven society, it is no surprise that data scientists are in high demand in industry. It is difficult for government-sponsored research projects to compete for data scientist professionals when they are highly sought after by industry, who can o er substantial compensation packages to recruit and retain talent. At the workshop, there was significant anecdotal evidence of loss of data science researchers to industry, where they are highly compensated and their contributions highly valued.

Computer Science and Mathematics Challenges

Research Challenges: 1

Collaboration

Large-scale scientific exploration in domains such as high-energy physics, fusion, materials, and climate involves large national- and international-scale collaborations, with considerable diversity in expertise, geographic location, and institutional affiliation. Thus collaboration is an essential ingredient of the "Big Science" that often produces "Big Data" [5, 6]. Increasingly, it is fundamental also to the science that is the lifeblood of many DOE experimental facilities, as the experiments performed at those facilities produce more data and become more interdisciplinary. Collaboration extends throughput the life cycle of the data and software artifacts that frequently form the focus and output of collaborative work, and touches on essentially all aspects of communication, computation, and data analysis.

Discussions during the workshop made it clear that as collaborations produce more data, existing data management, workflow management, and collaboration systems are hard pressed to keep pace. As discussed in previous DOE workshops on collaboration in science [7, 8, 9], R&D is needed to scale all aspects of the discovery process so that research can proceed rapidly and reliably despite bigger data, bigger teams, and more complex analyses. The 2011 report proposed, for example, an R&D program to create a deeply collaborative and collaboration-enhancing environment spanning the DOE laboratory system in which [7]:

- All data, code, and documents system-wide would be accessible, discoverable, reusable, reproducible, and computable (subject, of course, to access control).
- Those same information products would be linked by a distributed knowledge base that permits automated navigation of content and connections.
- Advanced software and computational processes would be available on demand and used routinely by every researcher.
- Collaboration would occur within spaces that people want to use even when they are not collaborating.
- Intrinsic and proactive security mechanisms would encourage rather than discourage collaboration, while protecting against attacks.
- These capabilities would be as intuitive, flexible, and collaborative as the best modern consumer software. Imagine if research data and software were as easily accessible as movies from Netflix and applications from an App Store.

The needs for such capabilities have only grown in the subsequent five years, and particularly for EOD, which place particularly challenging demands on collaboration technologies. We focus here on some aspects of these requirements that were emphasized at the September 2015 workshop that produced this report.

An important and repeated theme in this workshop was that EOD's unique status as an observation of the natural world can allow it to have value, and find uses, at times, at locations, and in disciplines distant from its original creation. EOD are frequently used by many scientists beyond its original developers, including many non-experts, and for purposes other than those for which it was originally created. Thus, tools and technologies to support collaboration around EOD must pay careful attention to preserving and indeed cultivating the life cycle of both the data itself and the various artifacts involved in its creation and transformation. Those artifacts can include code, workflows, and visualizations.

In the sections that follow, we first list some key findings concerning collaboration around EOD and then highlight new areas of R&D that need to be addressed by the DOE community to enable robust and e ective EOD-based discovery.

1.1 Findings

Importance of preservation. EOD is di erent than simulation data. Importantly, it cannot be regenerated. Its value often persists over time and it is frequently useful for people beyond its creator and for di erent purposes. As an observation of the natural world, it has a privileged truth value, and thus its accessibility and reliability are particularly important.

Importance of provenance and lifecycle. The unique properties of EOD can allow it to serve as the root of broad, deep, and long-lived collaboration networks. Thus, the data, code, and workflows that underpin EOD need to be accurately preserved, linked, and annotated, so that future collaborators can learn and build o EOD. By recording how EOD was used, this information can support future science, provide a basis for reproducibility, and permit learning from experience.

We must move from people finding data to data finding people/code/workflows. Exploding data volumes make current interaction and collaboration models based on people finding data untenable. When anything gets too large, one needs automation. Automation can usefully be based on the observation of past patterns. When a new datum is produced, it should be linked automatically to the people, programs, and workflows that are likely to find it useful.

EOS needs computing for control and steering in order to increase its overall value. Rapid response is needed for efficient instrument use, new computer-in-the-loop experimental modalities, and collaboration processes.

EOS engages user communities for whom the ease of use is critical. There are experimentalists who prioritize performance and computational scientists who care about ease of use. But in general, more experimentalists care about the ease of use and more computational scientists care about performance.

1.2 Data Lifecycle as a Basis for Collaboration

Collaborations are often structured around data, as when two experimentalists want to compare results; an experimentalist and a modeler want to use data to test a model or vice versa; or a new downstream researcher wants to use data for a new purpose. Inevitably, such collaborations also involve code, as it is rare that data can be understood or applied without understanding, and often developing new, software. Data, code, and workflows cannot be e ectively reused, and will often not be shared, without detailed information about the process by which they are first produced and then transformed over time. Such information is fundamental to collaboration: it provides data consumers with confidence in data's origins and data producers with confidence that their contributions will be recognized.

State of the art
The scientific community has made much progress on methods for capturing and representing metadata and provenance information. However, we are still far from the situation in which data captured for one purpose can easily be reused for another. Even routine use of data for an intended purpose is far from straightforward.

Challenges
A frequent obstacle to e ective EOD use is the need to understand "hidden information"—specifics of the codes or experimental apparatus used first to produce data and then to reduce that data (typically by large factors) to the data that is published and shared with others. Thus, for example, we find people having to "phone the beamline scientist" for knowledge about data before using it. The problem is that important information about the intention of the algorithm, data, etc., is often not placed in the knowledge base.

Another obstacle to EOD use is the obstacles that researchers frequently face when seeking to apply custom analyses. The process of making EOD available to the research community frequently involves sophisticated computational pipelines to clean, reconstruct, and transform raw data. ARM (see §13) and EMSL (see §11), for example, operate sophisticated data reconstruction and analysis pipelines on data feeds from instruments. However, these pipelines are relatively static and necessarily focused on the most common or urgent needs. It would be desirable for individuals and groups to be able to define and run new pipelines and queries: ideally, plugging them into existing frameworks rather than having to build new frameworks from scratch. Similarly, it should be easy for individuals and groups to share new analysis procedures and the derived data products that they generate.

R&D needed
Research which can come up with new methods to expose this information is necessary for larger scale collaboration. See also the discussions in Sections 9 and 10.

1.3 Discovery Engines

The data produced by a specialized scientific instrument (or, in some cases, supercomputer) represents a unique and expensive resource of value to many researchers. As data volumes grow, it becomes impractical (or at least inefficient) for each researcher to download that data for local analysis. Thus, we see the emergence of a new form of instrument: the storage, computing, data, and code required to allow community analysis of a large data set—a system for which we use the term "discovery engine." Many individuals and groups may work on a single discovery engine over a period of months or years, asking di erent questions, and producing tens or thousands of publications. The community needs to have access to the data sets and the analysis and visualization tools, along with all of the provenance needed to interpret, reproduce, and extend results.

State of the art
Successful discovery engines have been developed within a few disciplines and projects: see, for example, the Sloan Digital Sky Survey's SkyServer [10], the SEED system for microbial genomes [11, 12], the MG-RAST metagenomics server [13], and the Open Numerical Laboratory described in Section 20.

The Sloan Digital Sky Survey (SDSS),[1] for example, has collected imaging data for more than 35% of the sky with photometric observations of ~500 million objects and spectra for more than 3 million objects. Importantly, SDSS does not simply provide the community with access to raw data: the SkyServer[2] provides a range of interfaces for querying and accessing the data, including the CasJobs interface [14] for running

[1] http://www.sdss.org.
[2] http://skyserver.sdss.org.

computationally intensive SQL queries. As of March 2016, use of SDSS data has resulted in more than 5,800 refereed papers with greater than 45,000 citations.

The SEED[3] was first established in 2004, at a time when large numbers of sequenced bacterial genomes were being produced, with the goal of producing superior annotations (e.g., labeling genes with their functional role) for the first 1,000 sequenced genomes. To this end, the SEED team pioneered a new approach to genomes annotation based on the annotation of subsystems by expert annotators across many genomes. Users upload genomes the system for automated analysis and annotation; genes are called by comparison to the knowledge maintained within the SEED system. As of 2013, more than 12,000 users worldwide had annotated more than 60,000 distinct genomes. The related MG-RAST (the Metagenomics Rapid Annotation using Subsystem Technology) server,[4] launched in 2007, has as of March 2016 processed 239,314 meta genomes totaling greater than100 trillion base pairs for more than 12,000 users.

These usage data illustrate the impact that discovery engines can have on their communities. Importantly, the systems cited are all easily accessible by researchers with limited information technology experience and resources. They thus serve to both empower researchers who could not otherwise easily analyze data from new instruments, and as *loci* for collaboration around that data.

Challenges
The need for discovery engines arises in essentially every field of DOE science in which data volumes have become large: climate, materials, biology, cosmology, and many others. But significant challenges must be overcome before discovery engines can be constructed and used on a more routine basis. Some are listed in Section 20. A major cross-cutting challenge is that such systems are currently labors of love for their developers, developed and sustained with limited resources and with sometimes *ad-hoc* solutions to technical challenges such as resource management, data representation and curation, and data sharing and privacy.

R&D needed
New methods and tools are needed to streamline the process by which such systems are developed and sustained by individuals and collaborative groups, especially as growing data volumes and greater analytical complexity increase both demand for discovery engines and the costs associated with running them.

Increased computational demands leads to a need for scalable provisioning and policy-driven resource allocation, so that limited computational resources can be allocated e ectively within and across large collaboration groups. Cloud computing platforms seem well-suited for hosting discovery engines, although interestingly none of the systems listed above is currently hosted on commercial cloud services. More work is required to understand the computer architectures best suited for di erent classes of a discovery engine.

The power of a discovery engine derives from the quantity and quality of the data that it maintains. As new data is ingested and processed to generate new derived data, the discovery engine's knowledge base continuously evolves. So too will the analysis tools used to perform such analyses. Methods are needed for capturing such processes, so that results can be extended and collaborators can look at the graph of results and see how results build upon other data. Existing methods for recording provenance (see §9) likely need to be extended to enable the provenance of each data element to be determined and reasoned about.

Another important aspect of discovery is privacy. As collaborations grow, some information must inevitably be private for a certain period of time. (For example, the SEED and MG-RAST projects allow for private genomes and metagenomes.) Privacy policies need to be described so that di erent people in a collaboration can access more or less information as their roles change and data ages.

Both raw and derived data maintained by discovery engines needs to be accessible as objects. Users need to be able to query these objects and download small chunks of information from the original data. Data

[3] http://pubseed.theseed.org/.
[4] http://metagenomics.anl.gov.

may be used improperly if context is missing. Thus, the intentions of the tools and the data need to be adequately described (e.g., with appropriate ontologies) so that the accuracy of results can be understood and controlled.

1.4 Frictionless data movement and sharing

Collaboration requires communication, and collaboration around data often requires data movement. Distributed collaborative teams frequently want to move data from source to discovery engine, to archival storage, and/or to local computers for local analysis. In other cases, they want to move work to data. Anything that hinders easy, reliable, high-speed, and secure data exchange ultimately hinders collaboration and discovery.

State of the art
DOE investments in high-speed networks and data transfer technology have greatly accelerated data movement speeds over the past decade. The Globus service, in particular, is deployed at all major ASCR facilities and many experimental facilities (e.g., the APS; see §18) and is increasingly integrated into application work processes and data portals [15, 16]. However, many factors continue to hinder rapid data exchange, including local network, computer system, and storage system architectures that throttle end-to-end speeds; incompatible or baroque security systems that get in the way of remote access; disorganized and/or uncatalogued data that cannot be easily discovered; and data in unfamiliar or complex formats that cannot easily be interpreted.

Challenges
Many challenges must be overcome if we are to achieve the data fluidity required for e ective collaboration around large EOD. The following are examples identified during the workshop, with references to selected science projects in which they arose.

- Performance: As data sets grow in size, performance is often the key enabler, as collaboration cannot proceed e ectively if delayed by long data transfer times. Near-real-time access can permit qualitatively di erent, more interactive and collaborative, discovery modalities. This requirement arose across most science domains represented at the workshop: for example, APS (see §18), ALS (see §14), Climate (see §12), and ARM (see §13).
- Discovery: This is another major obstacle to e ective EOD use. Although there has been and continues to be much research in resource discovery, new research is needed to discover workflows, analysis and visualization code, papers, and data.

R&D needed
Workshop participants referred repeatedly to their desire for a "Dropbox for science," a technology that would allow large EOD to be managed and shared much as Dropbox provides for smaller data today, with the location being transparent, sharing straightforward, and synchronization automatic. Workshop participants were quick to note, however, that they need something more than just "Dropbox on steroids": they want technology that also provides for data indexing, discovery, and subsetting, for example, and that works e ectively on all computational resources, from the smallest to the largest. The realization of this goal, with all that it entails, can be a major focus for DOE research.

Work is required to improve data transfer speeds, with a particular focus on end-to-end performance (e.g., from the detector to the computer) and to/from the many smaller sites at which DOE researchers work.

1.5 Usability

EOD's potential for broad application makes *usability* an important concern for data and for the software, workflows, and other resources used to transform, communicate, and manipulate that data. This theme arose repeatedly during workshop discussions. Usability was viewed as essential if researchers are to work e ectively (and collaboratively) in a world of complex data, software, resources, and services, EOD is frequently used by researchers from other disciplines who may lack the computer skills possessed by the EOD's original creators. This situation arises, for example, in the case of ARM data (see §13), which may be consumed either in its raw form or (more typically) via various derived products by atmospheric, environmental, and other scientists.

The importance of usability is also discussed in Section 3 in the context of HPC software. However, no specific recommendations are made that speak to how to improve usability.

State of the art
User interface and human factor considerations have not traditionally played a big role in the design of advanced scientific software, which indeed is often viewed as requiring considerable expertise. Some groups reported that they have engaged user experience (UX) experts in their R&D teams, with positive results. However, they also noted that obtaining and sustaining funding for such people is not easy.

Challenges
Usability needs to become a focus of science software and data systems if we are to expand the scale and scope of collaboration, within and across institutions and disciplines.

Usability issues arise, in particular, when working with large and complex data, which is often difficult for communities outside the original creators to deal with. We need methods that allow users rapid access to simplified data sets, while also helping them to access other, more complex upstream data by themselves.

R&D needed
The usability challenge is broad and will require a sustained e ort across a wide range of areas. We expand here on two that arose in discussions, drawing also upon material from previous reports [7]:

- Identify management: The secure establishment of identity for di erent purposes and the management of the multiple identities that any individual inevitably possesses are fundamentally difficult problems that result in challenging usability concerns. Given the complexity of DOE science, progress will require both research and e ort applied to development and deployment.
- Cloud services: One reason for the poor usability of science software is its often bespoke nature. One answer to this problem is likely to be to get individual researchers, research teams, and laboratories out of the business of installing and operating the software and other information technology (IT) used in their research. In the consumer and commercial IT, this is a common practice: all sorts of software, and in particular collaboration and data management software, is routinely outsourced to cloud providers. The successful realization of such outsourcing approaches will require answers to a range of challenging research questions. What are the critical processes that underpin modern research—the equivalents for small and medium research teams of payroll, accounting, and customer relationship management for small and medium businesses? What are the foundational elements on which we can build robust, secure, and scalable research data management and collaboration solutions? How can these elements be integrated with supercomputer centers and other DOE facilities? How do we scale solutions to massive data, large teams, and high-throughput processes? What UX elements are important in research? (Companies such as Netflix, Google, Apple, and Amazon have pioneered approaches to consumer UX that have proved transformative in their usability. Will similar methods work for science?)

Research Challenges: 2

Mathematical Aspects of Data Analysis

The mathematical formulations and numerical algorithms for the analysis tasks used by DOE's facilities are as diverse as the science that these facilities enable. While these formulations and algorithms vary across the facilities and science applications, there are common features, as captured by the findings below.

- A common theme from all facility representatives at the EOD workshop was that data analysis is a major bottleneck to scientific discovery, and increasingly so.
- The current use of o -the-shelf methods for data analysis places deleterious limits on the fidelity, scale, and complexity of the analysis of experimental and observational data. Substantial opportunities exist for methods that account for the unique features of data analysis at DOE facilities, including the phenomena generating the data, the data acquisition, the storage and computational resources available for analysis, and the science questions driving the analysis.

Below we expand on six key topics, each identified by multiple use cases as being a critical need for scientific discovery. Although we highlight mathematical challenges, advances in these topics will require that related computer science challenges be addressed, as detailed in subsequent sections. In Section 2.7 we summarize crosscutting research in mathematics and data analysis needed in support of these topics.

2.1 Multimodal Analysis

Data acquisition frameworks in which data are acquired from di erent sources (e.g., di erent types of sensors, detectors, or simulations; under variable conditions; in multiple experiments) are often called multimodal. Analysis of multimodal data typically involves data fusion: deriving a single representation of the data obtained from multiple sources. The data fusion and integration process can introduce additional errors, as discussed in Section 2.6.

Characterization of data fusion involves a transformation (often hierarchical) between observed parameters and a decision or inference. Research on multisensor data fusion predates the advent of the so-called big data era and has been applied largely in areas such as automated target recognition, surveillance, guidance, and control [17].

The majority of the EOD workshop participants from the various science domains reported on applications

involving multimodal data, including data from neutron and X-ray scattering (§16.1.1), hyperspectral data from scanning probe and electron microscopies (§17.1.2), and data from new modalities enabled by the Advanced Photon Source upgrade (§18.2.3). Since multimodal analysis includes problems where experimental and observational data are combined with simulations, in many data assimilation use cases one form of data is used to improve understanding about another.

Challenges
The combination of multimodal data sources generates new degrees of freedom and thus requires more complex analysis than does exploiting each modality separately [18]. Incorporating data from multiple sources increases the volume and heterogeneity of the data being analyzed. Data from each new source brings its own level and type of accuracy. Furthermore, data from di erent sources can have complex interdependencies that must be accounted for in multimodal analysis.

R&D needed
Promising methods to address the cited challenges include matrix factorizations and tensor decomposition methods with constraints that are both convenient mathematically and physically plausible, pyramid-based data fusion schemes, nonlinear optimization for complex inverse problems, region-based fusion rules with statistical weighting, heterogeneous graphs [19], data and image registration techniques, and other heuristics. New, *adaptive* sampling, *adaptive* data assimilation, and *adaptive* experimental design techniques could maximize the degree of complementary information borne from di erent modalities (e.g., to close the loop between the Atmospheric & Radiation Measurement (ARM) facility's observational data and simulations; see §13.1.3). Methods for propagating information across scales can facilitate analysis of data covering disparate spatial and temporal scales. Also needed are methods that rigorously account for interdependencies among the data, and mathematical strategies to deal with missing data.

2.2 Uncertainty Quantification and Surrogates

Understanding and accounting for the uncertainty and error from various sources are critical to making useful inferences from the data collected during experiments and observations. These sources range from experimental and measurement error to uncertainty in model inputs, as well as variation due to the selection of samples and the choice of the theoretical model used to predict the quantity of interest.

The need for uncertainty quantification (UQ) is pervasive throughout climate applications (§12.1), as well as in high-energy physics (§21.4), to address questions such as "Did we see the Higgs boson?", "How fast is the universe expanding?", and "What confidence do we have in climate predictions from a model using observed data that has been assimilated into the model?"

Surrogate models are often used in UQ. These are simpler models, such as polynomials or neural networks, that are derived from data and models, and that can be used to provide predictions of the output values for a given input.

Challenges
Statistical techniques are often used to link the data to the inferences being sought [20, 21, 22]. However, this process can become challenging when the data is high-dimensional, large, and complex and represents multiple modalities. Building good surrogate models can be challenging in high-dimensional input spaces, when the amount of data available to build the model is small and the output is a complex function of the input. Furthermore, although Bayesian approaches are increasingly being adopted at DOE facilities, relatively little is understood about the sensitivity of these predictions to prior information (see, e.g., [23, 24]).

R&D needed
Research and development is required in UQ techniques that will connect data to scientific models for inference in data-intensive environments, where the variety, velocity, and veracity of the data are taken

into account in addition to mathematical and statistical considerations. Also needed is development of domain-specific statistical estimators and surrogates with mathematical guarantees such as consistency and optimality. Moreover, techniques are needed that quantify the sensitivity of analyses to domain-specific prior information.

2.3 Mathematical and Statistical Techniques to Address Data Quality

Understanding and improving data quality are important because the quality can a ect the error estimates associated with the data and the conclusions drawn from the data. Data quality is an especially important topic in cases such as the Environmental Molecular Sciences Laboratory (§11.2), where the complexity and heterogeneity of data, rather than the volume, represent the primary obstacle in obtaining deeper insights from the data.

Significant improvements have been made in the quality of data collected by instruments, as well as in signal and image processing algorithms that reduce the noise in the data [25]. Extracting linear relationships based on noisy data has been studied especially when the noise satisfies various independence properties [26] or when a model of the noise can be recovered (for example, in the case of denoising digital images [27]).

Challenges
Despite these improvements, the quality of data from experiments and observations can often pose a challenge to analysis. The data may have missing values due to an inoperable sensor, be distorted because of convolution by the point spread function of the detector (as in astronomy images) or because of detector jitter and vibrations, or be corrupted by extraneous objects (such as insects in the field of view of an imaging system used in atmospheric sciences). Furthermore, much of the experimental and observational data collected at DOE facilities exhibits structural correlations that violate standard noise-independence assumptions; depending on the type of analysis, these correlations can harm or ameliorate subsequent inference. The imputation of missing data [28] can also be challenging, especially for multimodal data.

R&D needed
At many facilities, algorithms are needed that can process large volumes of imperfect data in a timely manner. Promising approaches include using metadata such as detector event logs to enhance data analysis. Also needed are imputation, denoising, and registration techniques that take into account the domain-specific characteristics of the data generation and acquisition processes (and inherited noise) in order to detect and account for specific noise-creating events. Mathematical models of common sources of noise need to be developed that account for correlations among the data and sources of noise. In conjunction with these models, robust formulations of inverse problems are also needed to address data quality issues.

2.4 Dimension Reduction

Dimensionality reduction is viewed as a key technique at several DOE facilities. Needs in this area include dimension reduction with real-time feedback (§14.2), automating data reduction at the Spallation Neutron Source (SNS) for subsequent analysis (§16.1.2), and filtering to extract the desired material properties in scanning probe microscopy (§17.1.4). A particular form of dimension reduction is for exploring and visualizing high-dimensional data sets, a topic addressed in Section 4.5.

Challenges
Dimension-reduction techniques have been developed from diverse viewpoints, including linear algebra, signal-processing, statistics, and complex systems [29, 30]. Although numerous methods exist for reducing

the dimensionality of large-scale data sets, the results of black-box "workhorse" methods such as principal component analysis (PCA) or independent component analysis (ICA) are unsatisfactory for many scientists, such as users of the Center for Nanophase Materials Sciences (see §17). The physical constraints must be respected by the dimension-reduction methods, and the results should be interpretable from a scientific point of view. These aims are often accomplished by imposing additional structure on the output factor matrices (such as symmetry or non-negativity). Instead, PCA imposes artificial constraints such as orthogonality, which is not necessarily dictated by the underlying physics.

Although there exist dimension-reduction methods that can improve physical interpretability, such as non-negative matrix factorization (NMF) [31, 32] and CUR decomposition [33], robust implementations of these methods that can handle various kinds of large-scale data, possibly with missing or noisy entries, are not yet available. In contrast to PCA, these methods are also more expensive and cannot guarantee a global optimum solution. Arguably, one can also introduce additional constraints to the formulations of PCA, ICA, NMF, and CUR algorithms (see, e.g., [34]); adding such constraints can also result in formulations with unique solutions. Unfortunately, dimension-reduction problems typically become more difficult to solve when constraints are introduced.

Furthermore, by focusing on a two-dimensional (matrix) representation of data, the majority of dimension-reduction methods in use today ignore higher-dimensional (tensor) structure. Obtaining compressed representations of tensor data is an active area of research [35, 36]. Likewise, exploiting network/graph structure (e.g., as done by PageRank with great success in social and information networks [37, 38]) or other geometric representations [39] remains relatively unexplored for scientific data.

R&D needed
Research is needed on constrained dimension-reduction methods that address unique or particularly challenging features of data at DOE facilities, and that enable ready interpretation of the reduced results. Methods should be able to handle large data sets with missing values without explicitly imputing them, which would otherwise create memory problems. Such methods should, for example, be able to di erentiate between zero-valued entries and missing values that are potentially nonzero. Promising approaches could include using techniques similar to those found in sparse numerical linear algebra. Furthermore, algorithms are needed that can efficiently incorporate constraints that directly enforce desired physical properties. Also needed are efficient algorithms and implementations that can operate on tensors (multi-dimensional arrays) as efficiently as they do on matrices.

2.5 Streaming Data Analysis and Feature Tracking

Streaming algorithms perform online processing of data streams that are too large to fit into memory at once [40]. Consequently, analysis has to be done as data are being collected, and the algorithms can access only a small window of data.

Streaming data analysis shares features with in situ and in-transit methods. Current strategies to manage analyses efficiently and handle storage constraints properly include in situ calculation, in which diagnostics are performed on the fly and only summaries of information are archived. Many facilities, such as the expected LCLS-II upgrade (§15.2), foresee high-throughput environments where such streaming analysis is critical. Streaming analysis can also be used as a quality filter, such as in the case of online monitoring in order to stop taking measurements when a sample goes bad (§15.1), and for quality assurance and real-time reconstruction adjustment for protoDUNE (§19.1.5).

Feature tracking is a typical task in in situ analysis that targets the detection of key data attributes, often from high-resolution spatiotemporal data sets, continuously updating the resulting predictions. Examples of such tasks include tracking particles in the liquid argon time projection chamber (LATPC, §19.1.2) and tracking regions of interest in an extreme-scale simulation (§20.1.2).

Challenges
Many popular data processing, filtering, and tracking methods maintain a "sketch" of the underlying stream in order to be able to approximately answer queries. For example, Bloom filters [41] are useful for approximate membership queries with applications in biology, whereas algorithms for counting distinct elements, finding frequent items, and automatically finding regions of interest are helpful in astrophysics (§20.2) and genomics. Sampling-based methods also rely on maintaining an approximation of the data. One challenge in all such methods is not to lose important information during the process.

Furthermore, even for basic streaming operations (such as those performed during an experiment), typical facility users lack tools to reproduce these operations using slightly perturbed assumptions (§16.2). In order to tackle the volume of data, streaming data analysis algorithms generally need to be distributed, ideally with minimal communication requirements so as not to disrupt the dataflow. When used in conjunction with a simulation, in situ algorithms also must be faster than both the time required for a simulation time step and other streaming data sources from the physical experiment, or example.

Although one can recognize and track patterns in point-cloud data sets, such as those traditionally encountered in cosmology [42], approaches generally assume that the entire data set can be accessed simultaneously or in particular blocks. Performing such tasks for streaming data can be a challenge, depending on the manner in which points arrive in the stream. Tracking methods can be invaluable for studying dynamic processes, such as when computing displacement fields from automatically distributed landmarks [43]. Most approaches, however, are highly specialized or sensitive to noise in the data, which can result in many local extrema in the tracking objective function.

R&D needed
Research is needed on streaming-data methods that ensure a higher recall in order to avoid missing rare events. These rare events can be especially valuable artifacts in the scientific data collection process. Parallel libraries that provide basic data structures (e.g., Bloom filters, size estimation, frequent item queries) in an efficient manner on large-scale clusters are needed for the widespread adoption of those data structures by domain scientists. Development is also needed of tracking methods tailored for DOE facilities. Also needed are advances in modeling formulations or deterministic and stochastic nonlinear search methods that can overcome the many local minima present in current tracking tasks.

2.6 Data Acquisition

Modeling and incorporating characteristics from the experimental and observational data acquisition process are important in realizing the full value of the data. Both the process through which and environment within which data is acquired manifest themselves in measured data. For example, microscopes and samples can drift during longer measurements (§17), samples and detectors can heat up, and water can land on a sensor. DOE's light sources are just one area with an increasing demand from instrument sta for feedback between data analysis and the data acquisition process (§16.1.2); and proper accounting of background and run conditions is critical to DUNE being able to detect supernova bursts (§19.2).

Modeling the acquisition system is also a critical step toward performing optimal control. Control enables experiment steering, whether in conjunction with streaming analysis or simulation, a capability desired at many DOE facilities.

Challenges
When obvious, data that su ers from a change in environmental conditions (e.g., from rain or an unexpected occlusion) is often omitted from subsequent analysis. Particular forms of noise are often ignored; instead, techniques such as compressed sensing [44, 45] are employed as a general "robustification" practice. More recent techniques (such as when scan position errors are recovered in ptychography [46]) have

begun to exploit redundancy in noisy data in order to directly correct for e ects such as instrument jitter.

The noisiness of the data is particularly problematic when the data acquisition or fusion error is unknown, hence precluding the algorithms from providing confidence bounds. Approximate query processing (AQP) [47] can provide adequate answers to queries on noisy data by using sampling. The downside of AQP is that it requires k^2 additional samples in order to reduce the approximation error by a factor of k.

Combined data cleaning and sampling, whereby the knowledge obtained by data cleaning on the sampled part is used to improve the approximation error in the parts of the data that were not sampled, also has the potential to improve the answer quality [48]. However, the applicability of this approach to scientific data sets is an open question. Moreover, specific information about the data acquisition has been neglected thus far.

R&D needed
Current analyses would benefit from models that account for the acquisition process and environment in conjunction with the measured data and their use in data-based inference and reconstruction. Similarly, approaches that more accurately incorporate measurement statistics would particularly benefit areas that require shorter and shorter exposure times.

Incorporating specific knowledge about the acquisition process should improve "black-box" noise-reduction methods such as AQP; research is needed, however, to quantify the costs and benefits of incorporating various levels of information about the acquisition process. New mathematical abstractions could facilitate determining correct combinations of (acquisition-informed and generic) sampling and data cleaning for a particular instance of the data and analysis system.

2.7 Crosscutting Mathematics and Analysis Areas

Algorithms and approaches for addressing the six key topics discussed above share several common characteristics. Crosscutting research in the mathematics and analysis areas listed below will benefit several of these topics. Advances in these areas can also increase the range of analysis problems that today's methods can address, because such advances have the potential to make today's methods more robust, faster, and scalable to larger data sets.

Numerical simulations o er a "digital twin" to the physical phenomena and acquisition systems that underlie experimental and observational data. Data and simulations enjoy a symbiotic relationship: measured data can inform a simulation (e.g., through calibration and boundary/initial conditions), and simulations can fill in gaps—temporal, spatial, or otherwise—in the data. Furthermore, just like experimental and observational data, simulations can produce results at varying levels of fidelity. Critical research areas include scalable, multi-resolution simulations; techniques for data assimilation; and simulation-based design of experiments and experiment steering/control.

Techniques from machine learning, optimization, and statistics can play an important role in analysis: they can be used to improve the quality of the data, to enable multimodal analysis, and to build accurate surrogate models for UQ. Although these techniques have started making inroads into DOE applications, their application can sometimes be more an art than a science. In order to appropriately address the intricacies and subtleties associated with the use of these techniques so that the conclusions drawn from the data can be trusted, a closer collaboration between DOE domain scientists and analysis experts is needed.

Dynamics are an important factor in many of the processes that experimental and observational data seek to characterize. However, such dynamics are often one of the first features ignored (or coarsely

approximated) when faced with imprecise and incomplete data. Time series analysis, incorporation of dynamic models, and simulations of dynamical systems can all play a role in illuminating history-dependent and time-varying behavior in captured data.

Approximate methods are required when getting exact answers to analytical questions is too expensive. Moreover, when the data is noisy or incomplete, the concept of exactness is itself blurred. To get real-time answers to analytical queries, one has to resort to fast approximation methods. Versions of analysis algorithms that span the space of tradeoffs between speed and accuracy need to be developed to facilitate application on massive scientific data sets under different time constraints. Wherever possible, these algorithms should come with error bounds on the quality of the approximation.

Automation and abstractions of key analysis tasks can result in a better allocation of "humans in the loop" and thereby accelerate scientific discovery. A key hurdle to achieving this is formulating mathematical abstractions of the analysis process and/or underlying scientific phenomena. In particular, there is a need to formalize and quantify metrics that are currently qualitative (see §14.2, for example). Stochastic and statistical models, for example, can offer a formalism to notions of a "grey area" in many analysis tasks.

Performance and scalability are often a leading impediment specific to analyzing scientific data sets. While many data analysis toolkits exist in popular languages such as Python and R, their serial and interpreted nature makes them unusable on large data sets. The need for high-performance, parallel toolkits arose repeatedly at the workshop. In particular, domain scientists do not want to be required to stay abreast of the latest developments in HPC programming; instead, they would like to get higher performance from their very high-level languages. This allows them to think in terms of equations and mathematics, as opposed to programming concepts such as distributed objects.

Research Challenges: 3

Software Engineering and Software Infrastructure

Software is now one of the primary drivers in scientific research, being used at all stages of the process from data collection/generation, simulation, analysis, visualization, storage, and sharing. This increasingly important role means that we as a community must elevate the importance of software engineering and software infrastructure management. Here, we do not only refer to the traditional computer science definitions of software engineering, but we also embrace the extensions to deal with the extreme-scale science missions of cutting-edge and future DOE projects. Managing massive parallelism, scalable runtime engineering, and end-to-end workflow testing/validation, among many others, are all topics of importance for software engineering for DOE experimental, observational, and simulation science that go beyond traditional notions of testing and code validation. Scientific research will be accelerated by establishing best practices, creating sustainable communities around software, and rewarding the development of widely used software infrastructure just as we reward the publication of widely cited and impactful research.

Some previous e orts in this direction have faltered by focusing on a single, master system that all practitioners were supposed to (but did not) embrace. Instead, we must consider the best ways that the community can be served and encourage software reuse where practical without attempting to force such one-size fits all generic solutions. Scientific investigation is inherently diverse in its intent; software must have similar flexibility and customizability. Data size and complexity are increasing, as are the expectations of data preservation, reproducibility, and sharing of complete results. This means that it is our responsibility to establish, nurture and maintain sustainable software infrastructure using the best software engineering practices in order to accelerate the pace of discovery.

3.1 Findings

Software is playing a critical role in the acquisition, visualization, and processing of data. As such, it an essential part of the repertoire of scientific methods requiring established best practices to ensure high-quality and reproducibility. Moreover, as the essential facilitator between data and knowledge, software must be designed to support rapid customization, producing scientific tools that are easy to deploy and use.

- The curation and consolidation of software will lead to better scientific tools, requiring less investment and resulting in faster development times. Curation also necessitates building e ective dissemination sites, so that prospective users can rapidly find and acquire relevant software.

- Due to the increasing size and complexity of software, portable, high-performance computing tools must be developed to serve the broader research community. These development e orts must also consider software accessibility, making sure that these tools are usable in a variety of environments including, web, desktop, and HPC systems. In addition, data produced by the scientific community is becoming increasingly diverse and complex. This will require innovation in ways solutions are delivered to and shared across the community.
- Software must be refactored and designed for usability, reuse and customization. Such refactoring e orts also produce reproducible, higher quality systems. This approach supports domain-specific tools that are easier to access, use and deploy to specialized communities.
- In addition to data curation, we must consider software curation to enable both the longevity of data management and analysis but also facilitate the on-boarding of students and new research collaborators.
- Building strong communities supportive of essential software systems helps address the the long-term sustainability challenge. Communities must mix experts from computer science as well as domain experts and users.

3.2 Performance and Portability

State of the art
The state of the art in software performance and portability varies greatly across the experimental and observational data case studies. For example, the APS facility (§18) currently does not o er significant on-site computing or software resources. As a result, most software development is siloed into individual user's research groups, and the resulting software targets whatever local hardware is available, including local desktop-class machines. In contrast, the EMSL facility (§11) maintains its own software and HPC facilities for fully automated sample analysis. In between these extremes are many of the other cases, including the climate simulation and analysis (§13), which has pulled together a federated collection of tools developed by the larger DOE HPC community.

The experimental and observation data communities depend on a broad range of analysis software, including homegrown one-o solutions, complicated software frameworks such as Root, open-source community maintained frameworks such as Python and commercial software such Matlab and IDL. This diversity makes it especially challenging to leverage changes in hardware for improved performance while providing portability across multiple emerging architectures. There has been some attempts such as Numba for Python [49] and GPULib for IDL [50] but none of them achieved the wide usage and broad functionality needed by the community. The DOE visualization community has been developing VTK-m [51] with the aim of achieving performance and portability for many common HPC visualization tasks. Interoperability across many of these frameworks is either non-existent or in a very early state, making it currently very challenging to develop end-to-end solutions.

Challenges
Several of the facilities report exponential increases in the amount of data that future equipment will collect. Most case studies report an at least ten-fold increase in data within the next 10 years, and the APS (§18) expects their detector collection rate to out strip Moore's Law. This causes software issues both for managing the increased ingress of data, as well as the scaling issues with analyzing and interpreting the larger data sets.

Updating the software infrastructure to keep pace with the exponential data growth will be a challenge, one conflated by changing computer architectures as speed improvements have almost halted while core counts continue to increase. Keeping pace will require adopting new and/or additional computing resources, which will bring with them new software challenges. Transitioning from desktop-based serial processing

of data, as is still common in many of the use cases, will require both scaling up and scaling out. Scaling up is needed in order to make use of increasingly powerful heterogeneous processing environments that include accelerators like GPUs or Xeon Phis. Scaling out is needed in order to better make use of the parallel (HPC) and distributed (cloud) hardware investments DOE has made or is already planning. These changes cannot be made as bespoke, stovepipe-by-stovepipe rewrites to parallelize personal codes if the research community is to make full use of the expanding data.

There is a general need for HPC software to be portable. Some science collaborations, such as DUNE (§19), already directly address the need to migrate from dedicated compute resources built and managed at the local facility to leveraging shared resources at the DOE Leadership Class Facilities. As such the experimental facilities will have far less control over the construction of the hardware and will need to adapt the software accordingly. As the computer architecture can vary greatly across DOE facilities and between current and future generations, not to mention the variation in resource access for di erent parts of the research team, it is a high priority to manage code development so that it is portable and easily refactorable for future systems. Additionally, some science projects or facilities, such as scanning probe and electron microscopies (§17) and DUNE (§19), specifically call out the need to leverage di erent types of computing ecosystems that vary from traditional HPC systems to cloud computing resources.

Another challenge facing the experimental and observational data communities stem from the fact that the software in use is very diverse and much of it is homegrown using a large variety of frameworks. Much of this software is no longer maintained by their original authors—often graduate students who moved on to other things. It is unclear who is responsible of porting all of these analysis codes even if portability and performance objectives are met in newly developed software frameworks.

R&D needed
Many of the challenges facing experimental and observational data are shared with simulation and HPC research and development. However, care in evaluation and development of the existing HPC solutions so that they can deal with the distinctive features of experimental and observational data is critical. Robustness in the face of varying data quality, management of large vs. small updates, the relative importance of I/O, and other factors must be explored to transition existing solutions to broader adoption with community engagement. For portability, a great deal of progress has been made in designing HPC software that can run across many of the computer architectures that are expected to be used for experimental and observational data [52, 53, 54]. This existing work can be leveraged either by directly using the software or indirectly through software patterns learned from research. This consolidation of research and development can be beneficial to all those involved as is demonstrated with climate and simulation analysis (see §12.1).

There is also a deep need to develop processes that facilitate the creation of performant and future-proof software from the ground-up. Much of the analysis software targeting experimental and observational data is developed by graduate students and post-docs with limited experience in software engineering practices as well as a limited view of the long-term viability of the software they are developing. Furthermore, the adoption of proper practices and software frameworks is highly dependent on community outreach, community development and training. Therefore, holistic practices for sustainable, performant, portable and usable software need to be developed.

3.3 Usability and Accessibility

We believe usability and accessibility are important topics for enabling discovery from experimental data at DOE facilities. *Usability*, defined here as making software that users from novices to experts can use easily to interact with their data, is important because DOE facilities bring in new users at a high rate. Unlike DOE's simulation activities, where experts are trained over a period of years, DOE's facilities will sometimes work with users for a short period, and so the software they o er what must be both easy

enough to pick up quickly and powerful enough to do the customized analysis for their data. Accessibility, defined here as making data as well as necessary software available where scientists need to use it, is also important because DOE facilities have a wide range of users. Science teams may be composed of a variety of specialties, some of whom are strongly tied to the facility scientists but others of which may not be. All of these users have needs for access to community software for analysis and for tools to allow them to discover the community best practices and capabilities. Ideally these tools need to take a layered approach where simple uses can be learned quickly, and custom workflows can be automated or adapted to specific research studies.

State of the art

- Wide range usability. Some analysts consider Python as highly usable whereas certain science teams are used to graphical tools for analysis.
- Web interfaces and dashboards o er accessibility, depending on the nature of the security restrictions that are applied to the network. Some web-based tools are considered to be highly usable, while others have a reputation for being difficult to navigate and use.
- Adoption of many advanced algorithms (machine learning, image segmentation and registration etc.) into usable forms is currently poor.
- Accessibility of software is often limited to core science teams and sometimes only while on site (see §18 and §14). Communities that are not closely associated with the experiments develop their own infrastructure for analysis.

Challenges

- High-performance and usable software has been a challenge to achieve. The need to provide optimizations for high performance have a natural tendency to make the software specialized towards a particular, advanced community of users. Even if the code has the ability to cover many di erent scenarios with high performance, the parameters that a user must set to achieve that performance require a substantial additional investment of time, severely limiting the usability.
- Software specialized for experiments are often hard to find and learn. Since the analysis and data management software frequently has optimizations for science-specific or even particular instrument-specific requirements (the speed of data acquisition, specialized denoising, etc.), it is not uncommon that a user must locate a particular expert and his or her favorite analytics packages in order to get optimal use of the data.
- Accessibility also includes elements such as licensing, the right to reuse, extend, and share improvements—particularly for more commodity instruments.

R&D needed There is a need to better investigate the concerns and day-to-day usage patterns of a range of user types, from novices to domain specialists to instrument scientists. Quantification of the usability and accessibility of software is necessary for evaluation, diagnostic purposes, and constraints to future development processes. This necessarily requires developing metrics which capture the users' computing experience. Moreover the metrics must vary depending on the relative experience of the user (e.g., novice vs. expert).

Recommendations and best practices must take into account multiple levels of accessibility from basic access to software, through to the rights/access to reuse, extend, and share improved versions of the software. Software reuse can only be optimally achieved through the use of permissive open source licenses that encourage reuse, providing permission to share improvements and extensions. Ideally, this will facilitate the extension of software infrastructure beyond single institutions, promoting shared infrastructure through greater accessibility.

3.4 Building Software Stakeholders: Adoption, Refactoring, Reuse, and Community Resources

State of the art
Many software packages for the capture, analysis, visualization, and archiving of experimental and observational data end up being highly customized to the specific instrument and a specific investigator. Some of this is due to vendor proprietary software and formats used for the capture of the information, but much of it has to do with the *ad-hoc* nature of the creation of the software environment. There are some communities which have developed core packages, such as ROOT for high energy physics, but that is unusual. Many facilities and investigators have built their own environments to satisfy immediate needs for their science, but lacked the time, money or expertise to capitalize on those investments to share the results of their personal investment more broadly. This serves as a very high barrier for new users, or those who seek to develop new experimental capabilities at existing facilities. In addition to the previously discussed issues around the usability, portability, and performance of software, there is also a need to develop specific capabilities for developing and managing the community's engagement with a shared software environment. Much of that is currently done through human-driven processes (regular teleconferences, special conference sessions or birds-of-a-feather sessions on particular tools), but community support for tools to aid these processes is relatively underdeveloped currently.

Challenges
The highly varied nature of each experimental apparatus or line of investigation makes it very difficult to consider a single unified system (workflow, visualization, or otherwise) which might address all concerns, even within a relatively tight experimental community. Previous experiences with unified software environments have left experimentalists with software that was difficult to configure to the specific needs for their investigation. There is a desire for libraries and languages that can be easily manipulated or included by the investigator, rather than full environments. Conversely, engaging the experimental community from the computer science side can be difficult, as it can be difficult to come to a common terminology. The line between an individual investigator's needs, community requirements, and fundamental computer science R&D can be difficult to construct without specific investments to enable that exploration. Because simulation-driven data requirements are frequently easier to translate into a computer science framework, many of the tools and developments have been aimed at dealing with data management and analysis from those sorts of sources, rather than experimental and observational ones.

Experimentalists have become accustomed to stringing together multiple tools in order to accomplish their goals, and this leads to workflows that are error prone and complex. Funding has often focused on acquiring new experimental equipment with software often being an afterthought or left to the equipment vendors. The size and complexity of their data analysis is increasing, and these *ad-hoc* solutions are failing to scale. Furthermore, as reproducibility, data sharing, and peer review are increasingly important software tools are needed that provide facilities to save and share analysis pipelines that aid in replication, collaborative analysis, and wider sharing and publication of these artifacts.

R&D needed
Some research and development topics are as follows:

- Science-driven or software engineering approaches for refactoring existing solutions to be more widely usable.
- Community development infrastructure and support that enables experimental and observational scientists to meet, codesign solutions with computer science researchers, and e ectively disseminate expertise throughout the community.
- Support e ective methods for community outreach and engagement with other initiatives to ensure timely exposure and adoption of new technologies and best practices.

- Development of shared, scalable base-language capabilities (like SciPy does for uniprocessor analysis) with advanced features.
- Tools and environments to bridge new users into the data analysis, visualization, and management expectations of the facility.
- Development of data formats using best practices, with open specifications, and support from equipment vendors/translation capabilities from proprietary formats.

3.5 Sustainability

Challenges related to the development, deployment, and maintenance of reusable software for science are becoming a growing concern. Many scientists' research increasingly depends on the quality and availability of software upon which their works are built [55]. In particular, software quality requires formal software processes that ensure reproducibility and address concerns of verification and validation. Planning for software sustainability from the beginning will be particularly important if DOE plans to invest significant resources to the development of a software infrastructure for experimental and observational data management and analysis. Software sustainability will also be important to support data curation. Some of the data sets generated by the DOE community are expected to have a very long lifespan (more than years). Since this data is becoming larger and more complex and will require sophisticated software tools to make sense of it, it will be expected that such software has at least a comparable lifespan. "Packages will never see their full potential without user outreach, written guides, and worked-through examples/tutorials. As soon as maintenance and development of a package ceases, rigor mortis will soon set in" (see §18).

State of the art
The state of software sustainability varies widely based on the science domain, nature of the software and the particular experiment. For example, software for the management of data and workflows at the facilities is usually maintained and sometimes developed by the facilities. This type of software is usually maintained and improved over longer periods. On the other hand, in several scientific domains, software for data analysis is the responsibility of individual experiments or data analysis teams, usually consisting of small groups of scientists. These groups commonly use o -the-shelf open-source and commercial software components to put together an analysis infrastructure and this infrastructure has a limited lifespan, often that of the experiment itself. Additionally, while modern software processes have been adopted by some systems, in general there is large variation in their adoption and practice across the DOE.

Challenges
There are several major challenges to achieving the sustainability for data analysis software targeted at EOS:

- Science teams have a mission to do research not to maintain software long term. Funding cycles and the science mission for teams that produce a particular data set are usually shorter than the lifespan of the data set itself. This often means that maintenance of the software required for the analysis of the data ends before the data stops being useful.
- Teams are focused on their own science mission and coordination needed to create larger communities around software is hard for them. The challenges here include staffing and funding. Science teams are often overwhelmed by the need to prepare and run their experiment and do not have spare cycles for additional work that can make software sustainable.
- It is challenging to go to the next steps needed for sustainability which includes: documentation, testing, bug tracking, triage, maintaining message boards, outreach, etc. The software and hardware infrastructure required for these is often available but not necessarily easily configured and maintained by the science teams.

- Teams often lack expertise in advanced programming models, software engineering practices, development of portable solutions that can scale. The diversity of teams must be improved, establishing career tracks that encourage software engineers and domain specialists to work closely on scientific software, engaging with vendors or computer scientists in order to take advantage of new approaches, architectures, cloud and HPC resources.
- Additional time is required to generalize and harden solutions so that they can be reused, or to optimize their implementation for a number of use cases and/or hardware architectures. In addition it is difficult to obtain the funding for maintenance, refactoring, and/or modernization of software.

R&D needed

- Identify success stories for software sustainability. Both self-sustaining and externally guided/funded. There are various models for successful software sustainability. These need to be identified and studied by the science teams and software engineering experts to build a portfolio of templates that can be applied to DOE needs. One size does not fit all here.
- Develop and/or make available software infrastructure to help with processes necessary for sustainability. Documentation, testing, bug tracking, community message boards etc. Making it easy to establish processes that help community building and quality management are essential in software sustainability. These processes lower the barrier to contribute to the development and maintenance of software, helping the creation of self-sustaining software communities.
- Develop metrics to diagnose and evaluate the health of software systems, with particular emphasis on identifying systemic deficiencies a ecting the long-term sustainability of the software.
- Design scalable software processes that grow in sophistication as the size, complexity, and perceived value of software increases. The goal is to provide simple, easy-to-implement processes for smaller projects, which can be naturally extended as the user community and software system grows.
- Provide training and outreach on ways to build sustaining communities. It will be up to research teams that are building on existing software infrastructure to provide specialized data management and analysis capabilities to build in sustainability enabling processes from the beginning of the project. Initially, encouraging teams to opt-in to such processes and later providing training will be key to success.
- Develop processes to integrate software sustainability practices to new and emerging projects. In addition to community-based e orts towards software sustainability, it is important to encourage these practices when setting up new projects through programmatic processes. This may be similar to the way the National Science Foundation requires data plans in their projects for example.

Research Challenges: 4

Visual Data Exploration and Analysis

In this chapter, we provide a summary of the discussions surrounding the topics of visual data exploration and analysis (VDA). Algorithms, methods and tools in this domain have been reliably developed for successive generations of HPC platforms, so there is a strong foundation on which to build solutions for EOS needs. We note, however that as our case studies point out, each domain has specific challenges across its data workflows. Thus, successful VDA methods for EOS require both foundational and domain-specific R&D.

VDA is broadly used in two *modes*: interactive VDA, in which a scientist is actively querying, analyzing and visualizing data; and non-interactive VDA, in which computing is relied upon to drive data operations and create data products. We note that these modes have di erent thresholds for performance, latency and response, so both modes will require investment for EOS. A scientist sitting at a computer, waiting for query results or the rendering of a visualization, has a much smaller tolerance for delay than a computer running in batch mode. This human-in-the-loop interactivity, critical to enabling scientists to explore their data, will continue to challenge data workflows designed in ways that non-interactive VDA will not.

Finally, we note that a particular challenge for EOD VDA is the combination of experimental and observational data with simulated data in large ensembles of results. Each type of data has unique requirements, but combining and exploring them in concert—as a data ensemble for a specific domain—is a challenge unto itself.

4.1 Findings

- The design of visualization and analysis interactions, algorithms and methods with a human in the loop is a key element of successful scientific discovery.
- The integrated data exploration of experimental and simulation results present a new area of R&D, and it is critical to create novel algorithms, tools and methods to address this.
- The lack of domain-specific scalable visualization and analysis is a roadblock to science.

4.2 Scalability of Visualization and Analysis via Parallelism

Developing visualization and analysis approaches that scale with the size of available computing systems relies on solving two problems: decomposing existing serial algorithms into parallel tasks and mapping

that decomposition onto emerging architectures. Representatives from all three programs—BES, HEP, and BER—expressed the need to solve both problems.

State of the art

For example, Fermilab's ROOT data analysis framework [56] began in 1996 and still consists of several serial legacy algorithms that need to be redesigned to take advantage of shared- and distributed-memory computing resources. The Atmospheric Radiation Measurement (ARM) climate research facility processes data streams in serial mode and adds value to them to derive new data [57]. Synchrotron light sources likewise need to apply machine learning algorithms such as segmentation to images captured at beamlines. Early research in parallel segmentation for X-ray imaging has begun,[1] but it is applied to only one beamline out of over 100 at BES light source facilities. To date, no publicly available large-scale image processing capability exists.

Such algorithms must also be developed with an awareness of the machines on which they will run so that they can take advantage of emerging hardware such as many-core CPUs, GPUs, and nonvolatile memory (NVM). Otherwise, extreme-scale architectural characteristics such as high concurrency and heterogeneity will not improve performance, but will simply complicate the usability and portability. Few successful examples exist using architectural awareness to improve performance. Halo finding in cosmological data has been ported to GPUs on Oak Ridge's Titan machine [58]. Ptychographic reconstruction is now accelerated for several beamlines at the APS [59, 60, 61]. In the ACME project, several algorithms are in the process of being accelerated using GPUs [62]. These are individual success stories, but the majority of data-intensive codes used by experimental and observational facilities, while portable, are not designed or tuned to take advantage of the specific characteristics of emerging hardware.

Challenges

Developing scalable algorithms for processing experimental and observational data on emerging architectures presents several research challenges.

- *Data dependencies.* Parallel algorithms are limited by sections of the algorithm that must be serialized because of data dependencies. That is to say, not everything fits into a MapReduce [63] model where mappers and reducers are easy to identify. Particularly difficult to parallelize are algorithms whose data dependencies vary over time, and thus are not truly data parallel. Task-based fine-grain programming models [52] are one approach that shows promise, but for the most part, these programming models are still in their infancy and have not been tested on observational or experimental data-intensive tasks.
- *Communication.* Not all data analysis tasks are embarrassingly parallel. In fact, most require interprocess communication. While much work exists to develop design patterns for communication in analyzing simulation data [64], most of those patterns assume that, while not local to a processor, all data are available somewhere in the system simultaneously.
- *Mapping a problem decomposition to heterogeneous hardware.* Today, the problem decomposition and resource assignment steps in a parallel algorithm are statically defined when the algorithm is written. In general, the mapping is not dynamic or portable across di erent hardware configurations consisting of accelerators such as GPUs or Xeon Phi coprocessors. Libraries that abstract several different back-end devices from the programmer [54] can help, but again, they have not been tested in a streaming context.
- *Reducing data movement.* The high cost of data movement and the nearly constant I/O bandwidth projected for the next several generations of HPC hardware dictate that more visualization and analysis tasks be performed *in situ*. However, the computing resources collocated with an experimental apparatus may not be on the scale of supercomputers. How to compute or reduce data *in situ* us-

[1] Unpublished research at the Lawrence Berkeley National Laboratory's Computational Research and the Advanced Light Source divisions by O'Neil, Morozov, and Parkinson.

ing limited resources needs to be redefined, requiring new algorithms that operate in parallel but out-of-core [65]. Approximate algorithms may also be needed.

R&D needed
The above challenges can be met with a targeted investment in research and development of parallel algorithm development specifically for experimental and observational data. The following R&D e orts are needed.

- Applying parallel programming design patterns (out of core, task-based) to streaming data.
- Developing approximate (e.g., linear, sublinear sampling) parallel algorithms for high-volume, high-velocity, observational data that can have noise or measurement errors.
- Exploiting while abstracting hardware heterogeneity in parallel algorithms for computation and communication, and applying accelerated algorithms at a larger scale.

4.3 Data Ensembles and Uncertainty

A typical workflow data set for many use cases in this document contains data from instruments as well as data from simulations. Typically these include multiple runs of both types of data, and we call the collection of these data an ensemble. Ensembles of data require di erent algorithms, tools, and user interactions than a single run or experiment, and we note that as workflows become more complex, ensembles become the norm, instead of the exception. Closely linked with ensemble analysis is data uncertainty and variability, as workflows expand to include data about uncertainty which must be propagated throughout the entire workflow. Our workflows, algorithms and tools must be extended to work on collections of data sets that will include heterogeneous data, diverse spatio-temporal scales, sparse and missing data, and high dimensional data.

State of the art
The ALS case study (§14) notes that many "tools focus on single data sets of low dimension, so these data ensembles and high-dimensional data provide a particular challenge. New visualization methods must use novel visual encoding, interactive tools for dealing with higher dimensional data, and automatic algorithms to identify salient variables across ensembles or for dimension reduction with real-time feedback." Thus, ensembles and uncertainty are already part of these workflows, but current tools are not able to do the job today. The Open Numerical Laboratories (ONL) case study (§20) states that in order to perform UQ, tools and algorithms are needed to perform *ensemble access* to a potentially large number of simulations. In this mode, operations (such as averaging and comparison) can be performed on data from simulations with identical physics but di erent conditions or underlying components (such as random number generation methods).

In order to promote the integration of uncertainty in our workflows, we note the following science drivers.

- Synthetic diagnostics: firing sensors into simulation data to compare with experimental data;
- Techniques for data assimilation;
- Uncertainty visualization: understanding uncertainty—through mathematical modeling as well as through the visualization of the uncertainty—is a research area in its own right;
- Visualizing and analyzing variability: closely related to uncertainty, this concept reflects the amount of variation in a collection of data; and
- As part of understanding ensembles, tools and workflows must support the exploration of large parameter spaces that define the simulations, instruments, and other factors impacting the science.

Challenges

- Visual comparison: tools must span the spatial and temporal resolutions between data from experiments, sensors and computational models, and deal with representing data of di erent spatial and temporal scales. Along with this comes the challenge of how to e ectively analyze and visualize the two in combination.
- Visualization of ensembles of data, combining tightly integrated e orts in data management, access and visualization.
- All facilities are collecting complex sets of data from both experiments and simulations. Making data curation, access, and analysis a seamless process from collection to analysis is critical to all science areas. It is important to note that ensembles and uncertainty impact the entire workflow, as critical data must be propagated through the system and algorithms, capabilities and tools must all promote interaction with these uncertain sets of data.
- The BER ACME project frequently runs ensembles with "varying initial conditions, or internal model parameters to explore model internal variability, sensitivity to initial conditions, sensitivity of model response to process or parameter variations." These are "a class of uncertainty quantification for simulations of days to decades," and are critical to understanding the behavior of the complex climate system (see §12.1.3).

R&D needed
We note that ensemble analysis must be a tightly integrated e ort between the disciplines of data workflow, data management and visualization. This should include research in human-computer interactions, human-data interactions, UQ, high-dimensional data visualization and analysis. Research topics areas include:

- Visual representations of di erent scales and dimensions (in both space and time) in a useful and intuitive way.
- Multi-disciplinary teams to create the tools which combine statisticians, mathematicians, large-scale software experts and domain scientists.
- Efficient methods to impact simulations with a data feedback loop. Methods for *in situ* as well as batch execution are needed.
- For the BER ACME project, current strategies for managing (accessing, processing, and tracking) the large number of simulations (including ensembles of simulations used in UQ) are awkward, requiring a combination of manual intervention, to stratify di erent classes of simulations, and automated tools to track and analyze the consequences of systematic variations in parameter settings.

4.4 Data Reduction Algorithms Via Data Scalability

Data scalability encompasses data reduction, reconstruction, and compression of experimental measurements. All of the programs have acknowledged needs within this area. For instance, neutron scattering events may be summed into multiple pixels. X-ray science's data relies on tomographic reconstruction, and atmospheric radiation data streams need to be compressed until the raw data can be processed.

State of the art
The current state of the art includes tools developed by individual scientists through laboratory-supported tools [66] to tools supported within the community such as Pandora and Wire Cell for Liquid Argon (LAr) Detectors (DUNE LArTPC), and Bellerophon Environment for Analysis of Materials (BEAM), SPOT Suite, from the ALS. These tools may be used for all aspects of the data analysis process not just during the initial acquisitions phase.

Challenges
Developing algorithms for reducing, reconstructing, and compressing experimental measurements for new architectures presents several research challenges.

For instance, within the APS experimental data reduction is usually done either on beamline workstations or on a central cluster with nodes configured for and dedicated to specific APS beamlines, thus ensuring that hardware is always available on demand. Moving the computations to leadership class machines is problematic because of the latency as beamlines require computational resources within minutes or seconds and cannot abide with the queues employed on such machines.

Another challenge facing several programs is keeping data events as long as possible and only switching to reductions when needed (HEP and BER). This requires data compression, instead of reduction.

R&D needed
The above challenges, like other data intensive applications, would benefit from an increase in parallelism. This is especially true of those that make use of tomographic reconstruction which could utilize GPU technology. Specific areas would include:

- The application of GPU technology to the data acquisition and reconstruction stage.
- The development of scheduling algorithms for large HPC machines that allow for burst usage of streaming experimental data.
- The application of parallel algorithms to streaming experimental data.
- The research into techniques for variable binning and feature-based reduction.

4.5 Visualization and Exploration of High Dimensional Data

Many scientific applications routinely produce data sets that contain a large number of variables. Together with the spatial location and time associated with each data point, the total number of dimensions for a data set can be very high. This creates a great challenge to scientific data understanding because not only the size of data increases linearly, the complexity of data often grows exponentially as the number of dimensions increase. Furthermore, since our ability in spatial reasoning is limited to three-dimensions or even lower, visualizing high-dimensional data cannot be done easily. To address this issue, dimensionality reduction techniques are often used. However, for a specific feature of interest in a data set, the correlations between the variables, and the interplay between those variables' space and time properties are very complex, often unknown to the scientists. As a result, the e ective understanding of high-dimensional scientific data sets remains to be an unsolved problem.

State of the art
There are a number of visualization techniques for high dimensional data, and these break into three main categories. There are views and plots, such as parallel coordinate plots [67] and scatter plot matrices. Second, there are dimensionality reduction techniques such as multi-dimensional scaling (MDS) and principle component analysis (PCA). Third, interfaces have been developed that promote the interaction with multiple coordinated views of the same data. These include both general tools such as Tableau, which can present many views of data, and domain or task-specific tools such as Prism [68], which is designed for task-specific data.

Challenges

- It is difficult to extract the factors that contribute to the presence of a certain feature. It is even more challenging to understand the causality among the variables that contribute to a feature even if they are known.

- Because humans can only naturally perceive objects in lower dimensional space, visualization of high-dimensional data is often done by either visualizing one variable at a time, or juxtaposing multiple visualizations side by side.
- Encoding multiple variables in a picture through di erent encoding channels, mixing and matching the variables are mostly done through trial-and-error or based on prior knowledge.

R&D needed

- Develop novel visual encoding schemes for high-dimensional data so that a large number variables can be clearly visualized in one or a few images.
- There is also a need to develop interactive tools for exploring high-dimensional data space. To avoid trial and error and maximize the efficiency for data selection, automatic algorithms for identifying salient variables and values, and the relationship between them are very much needed.
- Design and develop interactive tools for dimensionality reduction with real-time feedback to the user so that an informed decision in the reduction can be made.

4.6 Interactive Data Exploration

For EOS, interactive data exploration will be of critical importance, focusing on the ways in which computing can impact and improve the way we interact with the complex data that result from experiments and simulations. This includes methods and algorithms for combining multiple data sources, query and search of large data collections, and the tools and algorithms needed to analyze them.

The rate of growth of measured data from powerful instruments and simulations is growing at a staggering rate, placing us in a world inundated with data, but oftentimes short on information and insight.

State of the art
Interactive visualization and analysis has been a focus of the scientific visualization and analysis community from its inception, resulting in a number of successful and widely adopted tools (e.g., ParaView, VisIt and Ensight) for wrangling with large scientific data.

To date, we have relied on interaction mechanisms with computers to dictate how we as humans interact with technology, and by default, data. However, this separation between human and data is making it difficult and time consuming for scientists and researchers to perform timely query and interrogation of increasingly larger data sets. A new paradigm, human data interaction, has been introduced by Haddadi, Mortier, McAuley and Crowcroft [69] to describe this area of research. The work has thus far been primarily adapted to social sciences and crowd sourcing applications, but we believe the concept of placing the human at the center of the scientific reasoning process has merit and much of the initial work could inform R&D in the scientific analysis community.

Challenges

- Due to a variety of system constraints that are expected to continue in the coming decade (I/O bottlenecks, memory allocations between simulation and analysis, etc.), there is a critical need for "[a]lternate strategies for analysis, including use of *in-situ* diagnostics and strategies for data compression" (§12.2). *In Situ* analysis is rapidly becoming a critical part of our analysis tool set, but novel strategies that address domain-specific opportunities for compression and optimized analysis are still a challenge.
- In particular, the BER ACME project notes that "[c]urrent strategies for managing ... the large number of simulations (including ensembles of simulations used in UQ) ... require 'manual intervention' [as well as] the use of automated tools" (§12.2).

- The BER ACME project notes that "data distribution is a bottleneck to climate science today that significantly impedes scientific progress" (§12.2). The ARM facility warns that "perhaps [approximately] 90% of data will soon fall into the category of only accessible for small case studies by experts" (§13.1.1). Thus, access to data access by collaborators across complex workflows is a high priority challenge. Integrated tools and interaction paradigms that simplify data access and distribution will impact all areas of science.
- Workflows may consist of many complex instruments, and their sheer complexity and problems, with increasing access to the scientists, is a difficult task. As noted by the ARM facility, these "are challenging to operate and produce large and complex data streams. Thus, many of these data streams remain unmined resources, and are quickly becoming the largest fraction of [the] data by volume" (§13.1.1).
- Better methods for integrating observational model data through improved retrievals and instrument simulations.

R&D needed
E ective VDA for EOD will require centering the design of interactions and methods with the human in the loop. While it is tempting to think about using similar interaction metaphors as those used in human computer interaction, a cursory examination reveals that in practice the metaphors will need to be examined closely and carefully thought out. In fact, the human (interaction, interrogation, query, reasoning) is placed at the center of the feedback loop and in between data and technology, requiring us to rethink our traditional visualization pipeline and feedback mechanism.

VDA for EOD includes the combination of data (from many sources) and the algorithms to analyze them. This will necessarily require the visualization community to re-examine how these sources are interconnected and perhaps adopt a more plug-in infrastructure for new and emerging data sources. With the human at the center of the reasoning and feedback loop, we must examine how interactivity can be achieved and scaled as the scale of the computing infrastructure scales and as the number of data sources scales. We have traditionally thought of reasoning loops having a single source of data that is consumed and then analyzed. However, we are seeing an increasing number of problems that require multiple heterogeneous data sources analytics support.

With that in mind, we have identified the following areas of R&D:

- New metaphors for interacting with data,
- Virtual experiments,
- Remote steering and operations of experiments,
- Query of the data and features within the data,
- Visualization linked to analysis and provenance,
- The co-design of sampling, compression and analysis pipelines for specific domains, and
- A task-driven development of data workflows that preserve scientific data and increase interactivity and collaboration.

Research Challenges: 5

Operating Systems, Runtime, and Architecture

In this chapter, we provide a summary of the discussions surrounding the topics of system software and HPC architecture. Today's system software stack was designed for supporting an environment in which supercomputer centers were largely ***cycle shops*** used by small numbers of highly specialized scientists. Moreover, HPC architecture is transitioning from an era in which performance improvements were achieved by simple frequency scaling to a much more complex environment. We have identified four broad categories of challenges: First, the influx of big data analytics requirements implies fundamental di erences in the usage model from an all-in-one cycle shop to a geographically distributed large-scale virtual computing center spanning multiple systems. Second, there is an emerging need for time sensitive computing. Third, EOS projects now require an expanded role of supercomputers. Fourth, the rapidly evolving HPC architectural landscape is bringing about notable changes.

Below we summarize our workshop findings, discuss the four contributing factors in more detail with an emphasis on the challenges they present, and provide our view of needed research and development.

5.1 Findings

The following points summarize the findings for operating systems, runtime and architecture research challenges surrounding EOS projects.

- The HPC ecosystem has become distinct from that found on more pervasive systems. HPC lacks adequate programming languages and programming environments to accommodate the needs of EOS scientists in a straightforward manner.
- Multi-system and multi-step workflows to accomplish big data analytics is not supported well by current system software stacks. For instance, the abstraction mechanisms available to EOS scientists to reflect their multi-system requirements are insufficient. This results in inadequate specifications between EOS scientists, workflow tools, and ultimately the system software. Moreover, workflows do not adequately incorporate emerging needs such as managing energy consumed or dynamically managing tradeo s for energy and resilience.
- Today's batch-oriented framework is too inflexible to support *ad hoc* requests by EOS researchers; furthermore, the single-center scope of much of the software limits big data workflows. EOS researchers

require the ability to augment special allocations such as high energy beam time with computed results that optimize experimental setups, but such usages are very sensitive to turnaround times of compute resources.

- Expanded supercomputer usage to buttress EOS projects is resulting in significant system software hurdles to efficiently employing HPC in support of DOE user facilities, particularly for novice users which are an important part of each community.
- HPC architectural changes have placed important new considerations on software. Recent shifts towards reduced memory capacity have far-reaching implications for applications, and emerging technologies will likely provide further shifts in recommended software approaches.

5.2 Making HPC Programming Languages and Programming Environments More Accessible

State of the Art
HPC environments place special requirements on programming languages and programming environments. For instance, the large processor counts found in leadership class machines compel certain characteristics in a programming language as well as a programming environment. Similar constraints arise from HPC's special needs in power efficiency and support for specialized hardware. As a result, the HPC ecosystem has become distinct from that found on more pervasive systems.

High-level languages and interactive tools with a user-productivity focus (e.g., Python, MatLab, UV-CDAT, Horace, Dave-Mslice and so forth) are commonly used by EOS researchers on workstations and departmental systems (see §12, §15, §16, §18). These languages and tools are used for rapid prototyping and interactive analysis; there is a desire to utilize them, only with scaled up data, on HPC platforms. However, most of these tools and languages have inherent design characteristics that prevent the high degree of scalability required for efficiently mapping to HPC platforms (e.g., they may be single-threaded or make extensive use of dynamic libraries). As a result, there is a productivity gap for HPC. The following quote taken from Section 18 describes the situation:

> Many scientists are very comfortable translating their data analysis concepts to computer code, but are most comfortable and productive doing so in an interpreted language environment like Python, Matlab or R, but not in Java or C++. At present, significant e ort from HPC experts is needed to adapt such codes to make e ective use of even the multi-core processors found.

The desire, therefore, is to do rapid prototyping and interactive analysis on expressive languages and tools, but at scale. There has been work (although limited) to address this.

One approach currently being pursued is the general-purpose language updated for HPC. An example is the Lua or Terra programming environment which o ers some of expressiveness of python, but without the baggage (the only external dependencies are a C compiler and 10,000 lines of code). This approach lets you do direct compiled code calls (which permits the performance of natively compiled and link directly to programming models like the Message Passing Interface (hereafter, MPI)), yet retains support for many of the desirable features of a scripting language [70], [71].

Another approach is domain specific languages (DSL). Examples of DSLs that have gained some traction in HPC include Scout (a DSL that targets e ective use of GPUs developed at LANL) and Liszt (a DSL for solving partial di erential equations on meshes developed at Stanford) [72], [73].

Challenges
New languages and tools face a chicken-and-egg problem at their beginning. Without sufficient support, researchers are reluctant to spend the time learning them. However, without sufficient user-base, new products are unlikely to garner the necessary support.

The languages and tools themselves face significant technical hurdles in achieving a technology that maps well to the HPC platforms of interest, provide sufficient expressiveness to be of interest to EOS researchers, and yield performance comparable to the best competitive technology.

Moreover, programming languages and tools touch upon two somewhat separate communities: the EOS research community and the HPC facilities community. Any successful approach must be mindful of both communities.

R&D needed

- Significant work is needed to bridge the gap between the expressiveness and ease-of-use of rapidly developing languages and tools, and the available languages and tools that map to the exascale systems.
- Methodologies to orchestrate computation across many di erent platforms (methodologies that allow optimal execution of workflows across di erent facilities and resources) are needed.
- Programming environments that integrate traditional analysis with simulation within the same environment are desired. Of particular interest would be language abstractions that allow just-in-time compilation for performance portability across facilities (if the job is ran at NERSC, do it one way, if on a di erent type of machine at the OLCF, refactor as needed there).

5.3 Enabling Resource Discovery, Marshaling, and Provisioning within Computing Centers

Spurred by the increased storage capacities and new policies which require that government sponsored research be made openly available, data re-use and curation for future uses is undergoing a dramatic shift. The resulting data-centric focus is leading to increased interest in big data analytics as noted by the Environmental Molecular Sciences Laboratory, Advanced Light Source studies, and Scanning Probe and Electron Microscopies (see §11, §14, §17).

Unfortunately, today's system software stack for supercomputers lacks the ability to automatically manage, schedule, and provision resources at di erent (tunable) levels of granularity. We would benefit from a metaphor or abstraction to convey our requirements, and ideally the new abstraction would be sufficiently expressive to convey all aspects for any range of systems or steps (i.e., any range of automation): we will refer to this idea as "automation via abstraction." This automation via abstraction is being driven by the needs of the domain scientists to achieve their end scientific result without being bogged down by the gymnastics of computer science to get there. Di erent aspects of the automation of importance to the domain scientists include scheduling and provisioning, data movement, workflow, and visualization. (From a computer science perspective, while the focus of automation via abstraction would be to deliver ease of use and performance automatically, this space would also enable the opportunity for computer scientists to automate and optimize along other dimensions of interest to DOE facilities, e.g., storage capacity, power, and energy consumption.)

As the landscape of HPC architecture adapts to new factors like power, energy and resilience, disruptive technologies are over turning our conventional notions for HPC software design [74]. Moreover, the goals, quantity, sizes, time scales, and diversity of workflows and respective software stacks make the description

of hardware and software requirements challenging. Many of the workflows appear to share common templates, but the parameters for resource requirements vary over several orders of magnitude.

State of the art
The relevant software systems are independent of each other and do not coordinate well. For example, domain scientists often write scripts to "glue together" di erent software (as part of an *ad hoc* workflow) or have to rewrite their code to take advantage of emerging technologies, e.g., GPU, burst bu ers, etc.

Although the community has various tools and capabilities for performance modeling of software, these capabilities have not been applied widely to scientific workflows for experimental data presented at the workshop. Many of scientific facilities have models of their workflows, but these models have di erent representations and levels of detail.

Challenges
The automation of any one of the following dimensions–scheduling and provisioning, data movement, workflows, or visualization–would be a monumental task. Doing so at varying levels of granularity only compounds the problem. The goal is to develop programming abstractions and runtime systems that can enable data discovery and data sharing between components of end-to-end workflows that integrate observations and simulations. The EOS community needs to help identify the appropriate abstraction interfaces.

The community needs to identify a common representation for modeling and profiling these workflows.

R&D needed

- Research is needed to develop system software abstractions, interfaces and mechanisms capable of supporting the advanced requirements stemming from big data analytics. In addition to the abstractions and interfaces, new enforcement mechanisms in system software may prove to be very beneficial.
- Research is needed to identify or create a common modeling representation that is capable of covering the majority of science use cases.
- The community will also need to create models of these workflows by investigating and profiling the existing workflows.

5.4 Time Sensitive Computing

We define *time sensitive computing* as the ability to obtain a specific type of resource through on-demand execution or through guarantees that a computation will finish by a certain deadline. This is driven by the desire to exercise critical experiment parameters during an inflexible access window in scenarios where computation is used to support an ongoing experiment, or when it is used to run a computational experiment', or finally when computation is used in evacuation and planning scenarios (see "Rise of the robot astronomers," "Real-time detection and rapid multi-wavelength follow-up observations of a highly subluminous type ii-p supernova from the Palomar transient factory survey" [75, 76] and §12, §13, §20). The requirement for time-sensitive computation may require HPC resources or may rely on non-compute resources such as I/O bandwidth. The time-sensitive computation may represent either analysis or simulation (as, for instance, in the "digital twin" and "computational experiment" examples). Note that time-sensitive computation may represent the need for an end-to-end integrated system that allows scientists to, for example, get weather data from instruments (and negotiate with those instruments), then analyze it, then transfer to Network Weather Service (NWS) within a certain deadline (§13). This implies the need for resource management methods including (a) preemption of a large parallel job in a "clean" way; (b) developing techniques for by deadline execution, i.e., true real-time execution, by improving predictability and modeling; and (c) focusing on response time when scheduling (e.g., such response time may be obtained

by targeting a result with lower accuracy or resolution). Finally, providing a controlled response time may involve the coordination of multiple resources including not just compute, but also storage facilities and potentially wide-area networks.

State of the art
Most scientific HPC data centers adopt a batch-scheduling model where a job can linger in the queue for an indeterminate period of time. Outside of small clusters dedicated to experiment support time sensitive computing is not available on DOE facilities.

Challenges
Existing schedulers and resource managers need to be extended to combine time-sensitive and batch scheduling modes in ways that both support the time-sensitive constraints and provide good utilization. Incentives need to be developed to support such extensions.

R&D needed

- Develop the ability to provide a true deadline capability in system software and interfaces.
- Develop improved supporting ability to coordinate storage facilities and potentially wide-area networks.
- Develop the ability to incorporate more sophisticated scheduling decisions based on extensible factors (e.g., power consumption, resource usage, and multi-step optimization).

5.5 Minimizing Barriers to EOS Use of Evolving Architectures

All science disciplines are experiencing rapid changes in the methods, machines, and infrastructure used for experiments. The multiple case studies detailed in the latter portion of this report give ample documentation of how leading scientific instruments and facilities are becoming more complex and more specialized. As leading-edge machines used for experiments become more complex and more costly, the competition for access heightens and the pressure to make the most of a given machine allocation becomes more intense. In fact, even slight improvements in machine setup and operational methods can have far-reaching implications for the scientists using the machines. As a result, using supercomputers in concert with real-time experiments to improve the e ectiveness of specialized facilities and machines is moving from a luxury to a necessity. Moreover, because the nature of research often dictates that graduate students or junior researchers are actually the hands-on people during much of the experiment, the use of supercomputers must present a shallow learning-curve for maximum e ectiveness.

Evolving architectures present special difficulties to EOS projects that rely on legacy software. The need to facilitate the adaptation (or porting) of applications to a variety of platforms can result in the costly need to re-write software.

Several science case studies have identified the need to plan for and use emerging technologies in their workflows such as object-based storage, NVM burst bu ers, and heterogeneous computing. These technologies could provide critical performance improvements in EOS scenarios, but they must be balanced against the costs of software development and other needs for the use case.

State of the art
Relevant libraries and software packages are specialized and some even proprietary. At the present time, these libraries and software packages cannot be expected to run across the supercomputing platforms of interest now and into the future. For example, high energy physics codes use CUDA to program the GPUs that massively accelerate their codes. However, CUDA codes only run on NVIDIA GPUs. Other strategies to adjust for rapidly changing computer architectures may involve the use of virtualization and containers

to run on multiple platforms. However, there still exists the need for a portable declarative language so that free code does not have to be rewritten as HPC architectures change.

Many di erent science cases are investigating the use of emerging technologies (e.g., object-based storage, NVM burst bu ers, and heterogeneous computing) that are specific to their workflow. Currently, however, there exists no coordinated e ort to explore and adopt these technologies across the EOS community.

Challenges
The EOS community needs to identify opportunities for emerging technologies in their existing workflows across science cases, which may be very challenging given the diversity and scale of workflows.

R&D needed

- Develop mechanisms that improve the supercomputing support for EOS projects. Make the facilities easy to use for inexperienced graduate students and junior researchers who may be involved in running experiments.
- Investigate emerging technologies with a high potential to impact these workflows. For example, provide domain scientists with sufficient information so that they can decipher opportunities for emerging technologies in EOS workflows.
- Develop the ability to cooperate simulations and EOS projects.

5.6 HPC Architectures

Many of these new capabilities for experimental data are set in a time of rapid change for HPC computer architectures and software [74, 77, 78, 79]. First, architectures are growing more complex in response to the constraints of power, cost, and reliability. Second, system software, programming models, and runtime systems are changing to accommodate these architectural changes. In particular, the I/O subsystems for HPC architectures are changing rapidly to accommodate the plateauing bandwidth to external resources like filesystems and networks.

In particular, constraints on main memory capacity and external I/O of future extreme-scale systems have several important implications for experimental data workloads. Most importantly, a limited main memory capacity throws the system architecture out of balance, impacting other system parameters and efficiencies. That is, in most experimental data applications, a smaller main memory per node reduces the amount of computation a node can execute without internode communication, increases the frequency of communication, and reduces the sizes of the respective messages communicated. All of these factors are known to lower application efficiencies; this relatively small memory capacity will be efficient for only the most computationally intense workloads.

Moreover, a critical trend over the past decade has been the removal of hard-disk drives from the node designs of large-scale supercomputers. Virtually none of today's large scale systems have mechanical hard disks physically present in the node, limiting working data set size to the size of the DRAM main memory. Typically, contemporary systems forward all parallel I/O requests over the interconnection network to specialized nodes that connect directly to a storage area network, typically using Infiniband links and commodity storage targets to retrieve or store the data as requested by the application. In some systems, a small amount of main memory can be reserved as a RAM Disk for the operating system and application use. An important consequence of this configuration is that these systems cannot support any demand paging of virtual memory as is typical in most other computing systems. As such, HPC applications are designed so that all application data explicitly fits well within the size of main memory capacity, while allowing space for other system functionality, such as those needed for the operating system and message passing runtime systems.

ASCR Computing Upgrades At a Glance

System attributes	NERSC Now	OLCF Now	ALCF Now	NERSC Upgrade	OLCF Upgrade	ALCF Upgrades	
Name Planned Installation	**Edison**	**TITAN**	**MIRA**	**Cori 2016**	**Summit 2017-2018**	**Theta 2016**	**Aurora 2018-2019**
System peak (PF)	2.6	27	10	> 30	150	>8.5	180
Peak Power (MW)	2	9	4.8	< 3.7	10	1.7	13
Total system memory	357 TB	710TB	768TB	~1 PB DDR4 + High Bandwidth Memory (HBM)+1.5PB persistent memory	> 1.74 PB DDR4 + HBM + 2.8 PB persistent memory	>480 TB DDR4 + High Bandwidth Memory (HBM)	> 7 PB High Bandwidth On-Package Memory Local Memory and Persistent Memory
Node performance (TF)	0.460	1.452	0.204	> 3	> 40	> 3	> 17 times Mira
Node processors	Intel Ivy Bridge	AMD Opteron Nvidia Kepler	64-bit PowerPC A2	Intel Knights Landing many core CPUs Intel Haswell CPU in data partition	Multiple IBM Power9 CPUs & multiple Nvidia Voltas GPUS	Intel Knights Landing Xeon Phi many core CPUs	Knights Hill Xeon Phi many core CPUs
System size (nodes)	5,600 nodes	18,688 nodes	49,152	9,300 nodes 1,900 nodes in data partition	~3,500 nodes	>2,500 nodes	>50,000 nodes
System Interconnect	Aries	Gemini	5D Torus	Aries	Dual Rail EDR-IB	Aries	2nd Generation Intel Omni-Path Architecture
File System	7.6 PB 168 GB/s, Lustre®	32 PB 1 TB/s, Lustre®	26 PB 300 GB/s GPFS™	28 PB 744 GB/s Lustre®	120 PB 1 TB/s GPFS™	10PB, 210 GB/s Lustre initial	150 PB 1 TB/s Lustre®

U.S. DEPARTMENT OF ENERGY

ASCAC December 9, 2015 24

Table 5.1: Contemporary ASCR HPC architectures. Source: Dr. Steve Binkley, ASCAC Meeting, Dec 9, 2015.

In this regard, recent trends in the research, development, and manufacturing of NVM technologies provide optimism that these technologies can be used to reverse this trend toward very small main memory capacities in these future systems. Research in new devices, such as Phase Change RAM (PCRAM) and resistive RAM (ReRAM), and innovative integration strategies could provide alternative memory designs that will have a broad impact on future extreme-scale architectures and applications.

Research Challenges: 6

Service Facilities

The Office of Science supports its scientific research with a collection of "service" facilities that provide computing and networking services for the science community. However, each science community or discipline has di erent workflows, analysis, and simulation requirements. How can these service facilities improve their ability to support science?

6.1 Findings

There are a collection of problems in technology, policy, and operating procedures that impede the use of SC supercomputer facilities by the data-intensive or experiment-based science communities. The two primary issues that need to be dealt with are the current policies and the fundamental design technology of supercomputing systems. The technology and operating issues can probably be addressed. Some of the policy issues have persisted for decades and would take concerted and focused work by DOE SC to be changed.

There was a general agreement that having access to supercomputer-class facilities would greatly enhance the development and use of next-generation instruments. These new instruments are expected to need high-performance computational capabilities only episodically and thus it seems more practical to use existing supercomputing centers rather than to build specialized systems. Unfortunately, the current architecture and operations of supercomputers were not designed with the needs of experiments in mind, and domain scientists need increasing levels of support to use emerging storage, workflow, and compute architectures to process increasing amounts of data.

Many of the case studies identified supercomputer usage problems related to data ingest, retention, data management, I/O from external storage and coordinating instrument access to supercomputers. This reflects the current batch-oriented, highly parallelized, maximal floating point operations per second focus of these systems. If supercomputing systems are to make a significant impact on the processing, analysis and visualization of large scale instrument data sets, they will need to implement new designs specialized for data processing, in which data can be rapidly ingested, processed and fed back to researchers. Alternately new hybrid systems could be designed that can accommodate both traditional and new users.

The issues related to policies include access to computational resources in a persistent community environment, and problems with heterogeneous user environments across centers. In particular, the lack of a common authentication scheme allowing single sign-on across centers was seen as a significant problem in the community use of supercomputing centers.

6.2 Accommodating High-performance Storage and Data Needs at Supercomputing Facilities

A recurrent theme from both the experiments and parts of the simulation community was that the supercomputing centers need to provide much more online storage whose retention policy can be determined by the users: for (real-time) collection, archiving, and the redistribution of data from instruments; for allowing community access to very large curated databases for query and download, and for the selection of data for supercomputer simulation runs; and for keeping enough data available to do multi-model comparisons.

State of the art
Data systems at HPC centers generally fall into two categories: 1) expensive, highly parallelized filesystems that are optimized for streaming large volumes of data on and o compute nodes at maximum transfer rates, and 2) large-capacity archive systems that are optimized for storing large amounts of data at a minimum cost and are able to move data on and o the aforementioned filesystems at useful rates. Neither was designed for quick random access of data that remains resident for long periods of time, and both have poor performance characteristics for random access or high numbers of I/O operations per second.

The state of the art for streaming data to high-speed storage from supercomputers consists of various disk arrays, or increasingly NVRAM-based systems, that are accessible to the compute nodes over a local high-speed network. This storage is designed to be accessed by jobs running on supercomputers and is typically configured as form of ephemeral ("scratch") storage. That is, users can put files into this storage but there is no guarantee that the files will remain beyond the lifetime of the job that is actively using them. This is typically implemented by various "purge" policies that ensure that executing jobs can stage all of the files that they need for execution.

The longer term storage is generally enabled via "project" filesystems and archived to tape stores (such as HPSS), but files located in this archival storage have to be staged to high-speed storage (e.g., on disk) for performant access by supercomputer jobs. Managing the location and movement of data through this hierarchy is generally left to the user, and is typically done with di erent tools at di erent centers.

Support is limited in current parallel filesystems (such as GPFS and Lustre) for fast searching of file metadata, or the attaching of user-defined metadata for extracting specific subsets of data. Limited use can, and has, been made of the GPFS policy-engine and Lustre's Robinhood system [80], but this has not allowed for tracking across filesystems at a supercomputing center. None of these filesystems o ers robust data management features.

More comprehensive data-management systems have been developed that include iRODS [81] and SRM [82, 83]. The former has achieved quite a widespread adoption while the LHC community heavily uses the latter. However, these solutions are generally quite "heavyweight" for both users and administrators and restrict the ways in which files can be accessed, requiring it to be through the data-management service (for example, PanDA [84]).

Challenges

- Current DOE supercomputing centers were not designed for the real-time analysis of experimental data and would require significant modifications if they were to take on this role. For example, up to terabit-per-second access would be needed for some facilities to store and process experimental data streams in real-time and do online data analysis. This is currently beyond current technical capabilities.

  Data centers providing resources for data intensive computing could help offload experiment facilities like LCLS that are not currently able to build and operate large-scale computing and storage systems. However, the design and configuration of these centers would need to change to meet that

specific need. As mentioned in Section 15.4, for example, general support for LCLS-II offline analysis alone would require approximately 100 PB of tape storage, a dedicated 20–100 PB of disk storage and a processing farm in 0.2–1 petaFLOP range with an aggregate throughput to the storage above 10 GB/s per PB.

Other key requirements are the ability for users to manage their data through the science facility (e.g., LCLS) tools and workflows and, ideally, to be able to use that facilities user-account (or a federated account) when accessing the data.

- HPC can be thought of as an experimental instrument in its own right. Soon the largest simulations will exceed several petabytes and such simulations will allow direct comparison with experiments.
  - There is insufficient tertiary storage cache to allow efficient running of such jobs.
  - There are insufficient lightweight data management tools to allow the migration of data from larger pools of storage for running jobs or for further user analysis.
  - There is a need for tools to process and resample data so that the output of experiments can be directly compared to simulations.
  - There is a need for tools to expose data to a wider science community to do experiments through analysis and queries of data. This might mean running experiments *in situ* (on the supercomputer with local data accessed during execution) or it might involve the migration of data to other filesystems or o site.
- Today's supercomputers are not designed for analyzing large data sets. For large-scale data analysis there are I/O choke points resulting in I/O time exceeding analysis time, which causes problems with scheduling. That is, there is a major mismatch between compute power and accessible storage. This relates to the progressive scaling of both computing and storage capacity: it is difficult to design a computing center in a way that experiments can point their data streams at a supercomputer center and require storage and scalable analysis on the data.
- There are insufficient mechanisms at facilities to share or disseminate data to satisfy DOE guidance and mandates. In addition, there are insufficient data policies that clearly define which data sets need to be shared and disseminated.

R&D needed
Lightweight, easy-to-maintain and high performing solutions for the challenges noted above must be deployed at facilities, covering such areas as listed below. The R&D required to develop *new* technologies in this area is largely covered in Section 8. However facilities also have a role in deploying and configuring both existing and new technologies to meet these community needs. Areas include:

- Mechanisms to define per-user policies and automate data movement between di erent tiers of storage or di erent storage systems.
- Improved filesystems that are optimized for both metadata access and large-scale data I/O.
- Methods for surveying the contents of the storage system for tracking a science project's usage and file locations.
- Methods for enabling user-defined metadata.
- Methods for querying or performing analysis on data *in situ* on supercomputing filesystems.
- Methods for optimizing performance of I/O systems that scale well.

6.3 Supercomputer Centers for Real-time Analysis of Instrument Data Streams

The ability to use supercomputers in experimental science is becoming a critical need. Instruments are getting more sophisticated and expensive, and this makes increasing the productivity of the instrument more urgent. Further, the amount of data coming out of modern instruments is vastly more than the previous generation of instruments, and that data is frequently much more difficult to analyze. This means that many instruments are operated "blind" unless a quick analysis of the data can give the scientist insight into how the experiment is progressing. The need for fast analysis of complex data using complex models frequently requires supercomputing capabilities.

However, the total real time feedback needed for such computation is typically relatively small compared to the lengthy process of setting up and operating the experiments. This implies that a highly capable system that is shared—like a supercomputer—can best fill this need as it is not economical to build a dedicated system for this purpose. However, jobs on supercomputers are typically scheduled far in advance and providing rapid, episodic access to instrument data is not usually practical. Thus, either new approaches to scheduling or di erent supercomputer architectures need to be developed.

State of the art
There are a few emerging examples of using supercomputer analysis of instrument data streams in real-time. Recently, near-real-time data from the Center for Nanophase Materials Science has been analyzed on OLCF's Titan supercomputer [85] by pipelining analysis first in ORNL's CADES data analytics environment. In a more store-and-forward approach, data from the ATLAS experiment has been analyzed in an opportunistic manner on Titan and NERSC. In these modes of execution, data streaming from the instruments are transferred to a staging area "adjacent" to the supercomputer, and innovative mechanisms are used to run the job. This method of staged or pipelined execution has been necessary because of both the manner in which experimental facilities collect their data, and the way in which data is made available to queued execution jobs on the supercomputer.

Recently NERSC has setup a "real-time" queue on its new Cori supercomputer to begin to address the real-time analysis needs. It relies on a small number of dedicated compute nodes to o er responsiveness, but also allows jobs to take priority on other resources once those are filled. It is also possible to preempt 'killable' jobs on these other resources. However this is built on the existing features of the Simple Linux Utility for Resource Management (SLURM) batch system and new technologies may be required to be responsive at large scale while maintaining the efficient use of resources.

Challenges
The main challenge of using supercomputers to analyze instrument data in real time is that there is currently no e ective mechanism to do preemptive scheduling of this type of work. Even if jobs could be co-scheduled, there may be I/O limitations in importing large-scale data streams for processing.

Beamline instruments, for example, need to provide feedback to experimenters as to how their measurements are proceeding while the experiment is in progress. Being able to spot a minor problem in the experiment early can spell the di erence between a successful outcome and a wasted week of beamtime, scientist time, and associated travel costs. Unfortunately, having to do on-the-fly analysis via a supercomputer queue means that measurements would be conducted "in the dark" because the job will not complete before the experiment does. Experimental scientists need computer systems that provide fast throughput and rapid access. Delays on the scale of hours can not be tolerated in this type of work.

To date, supercomputing user facilities have not been utilized to any significant level in the routine operations of beamlines, such as the APS. The key to changing this is in establishing mechanisms that allow beamline computing to run concurrently with the long-running batch jobs that represent the main workload for these large machines. The most e ective beamline computing should be scaled to use the largest amount of resources that can be deployed e ectively to provide a result to a user within minutes, if not

seconds, of the completion of the measurement. This means that the ideal use cases (when parallelization is possible) will employ large numbers of processors for short periods, with potentially long delays between tasks while the next set of data are collected. By design, these computations will optimally use only a small fraction of a facility's total capacity. Thus, a system dedicated to experiments alone is not practical (economically or operationally) making shared use of such a computing resource a high priority. How to schedule such jobs or have them run concurrently with batch jobs is a major challenge.

In addition to the scheduling considerations, there are programming language and resource allocation obstacles in establishing seamless workflows across diverse hardware and software systems that span observational instruments, and analytics resources. These systems often don't have e ective middleware that can interoperate and allow a functional end-to-end overlay. The lack of a sensing and messaging middleware that can be deployed programmatically is a continuing challenge.

R&D needed
The R&D needed includes:

- A better understanding of preemptive and real-time (hard and soft) reservation and scheduling mechanisms, and their interaction with the interconnect and filesystems.
- Designing data pipelines that straddle acquisition, transfer, staging, and ingest into the computing platforms. This needs to be constructed in way that closes the feedback loop—back to the instrument.
- Creating "first-class" workflow constructs that can be intuitively stood up and deployed as needed from observation instruments to supercomputing resources.

6.4 Federated Uses of Computing, Science and Network Facilities

There are emerging problems where simultaneous, or nearly simultaneous, use of multiple supercomputers is necessary. This could be enabled by federating a set of service facilities.

State of the art
Distributed computing resources have been used in federated ways for particular science projects and industry for some time. Some examples include the Earth System Grid Federation, the Worldwide LHC Computing Grid and commercial clouds such as Amazon EC2 or the Google Compute Engine. Recently the LHC community has also explored a more flexible use of compute resources, including cloud and supercomputers, as well as a more transparent use of storage resources, through dynamic federations [86, 87]. However, there is little general-purpose tooling that allows for the federated use of the supercomputing facilities.

One barrier to such use is a lack of common job control language across the supercomputing centers. Another is di erences in the operating system environment. This later issue has begun to be addressed by the increasing use of container technologies (such as Docker [88]) that might possibly be a common capability across the centers. For example, ACLF has an active project for testing Docker containers [89] and CADES has the "Cosmology INCITE project" [90].

Furthermore, NERSC is using a Docker-like tool developed in-house called "Shifter" [91] which is now enabling code developed at science facilities such as the LCLS and LHC experiments to be run directly on large NERSC machines, in a di erent OS environment than the base system, with minimal modification.

How the commercial cloud computing capabilities may complement or supplant certain supercomputer center services is a growing question. While commercial o erings are generally cost-prohibitive because of the technical needs for the highest performance and largest-scale data movement, certain public cloud o erings are practical for some applications such as burst-driven utility computing. The supercomputer

center of the future might consider hybrid strategies to o er the best complementary computing capabilities and a higher-level coordination with the private and public computing ecosystem.

Challenges
It would ultimately be desirable to be able to use the ASCR computing facilities with the same workflows running on all three and passing data back and forth during execution. However, the lack of a common language to define the job pipeline operations across centers makes moving jobs difficult. There is also no mechanism for co-scheduling jobs running at di erent facilities that need to interact during execution, and a lack of federated storage. Thus, it is not currently possible to use the ASCR centers in this way. Currently many experimental groups do make use of di erent systems, but it is manual and labor intensive. Challenges include:

- Incompatible access control requiring, for example, di erent security keys at di erent sites increases the barrier to using multiple supercomputing systems in a workflow.
- The lack of a federated approach to data storage and analysis resources across supercomputing centers and participating sites adds complexity and inefficiency to multi-center use.
- The lack of fault tolerance in computing centers compared to that found in cloud resources.
- Di ering operating systems and environments require the recompilation and, in some cases, di erent software to run at multiple sites.

R&D needed
The following areas of development could help scientists make use of multiple centers:

- Policy decisions or suitable technologies to allow scientists to use diverse resources with credentials from their science facility.
- Mechanisms for creating, establishing and scheduling workflows across computing centers.
- Mechanisms and deployment of technologies that allow for federation of storage resources at di erent facilities.
- Mechanisms of inherent failure tolerance in federated supercomputing centers. Certain experiments would benefit from high-availability supercomputing and data-processing. A federated and failover approach would address that need.
- Container technology available on large compute systems at supercomputing facilities. These should also be compatible systems, allowing use of common containers and repositories.

6.5 High-speed Data Movement Between Geographically Distributed Systems

State of the art
The ability to move very large volumes of data at high transmission speeds between geographically distributed systems relies on several capabilities: 1) a sophisticated network tailored for large-scale data flows for transport, 2) an infrastructure located at the sites (usually a national laboratory and DOE facilities such as the leadership computing facilities) built to deal with the ingress and egress of large data flows, and 3) the appropriate data transfer tools for large-scale data volumes.

Wide-area network transport
The wide-area network (WAN) must have the transport capability, both in terms of the transport speed and capacity to handle petabyte-scale data between the science facilities and scientists. DOE's Energy Sciences Network (ESnet) is the wide-area network that provides these capabilities for scientific research across

the United States and internationally into Europe. ESnet's network infrastructure consists of an optical system capable of 40x100Gbps channels, and a routed infrastructure consisting of one to two 100Gbps across ESnet. Most of the SC Labs have optical fiber access to ESnet with at least a 100Gbps link and the potential to provision more 100Gbps connections. A handful of large U.S. universities also have 100Gbps connections in places where ESnet can connect to them (e.g., Chicago, New York, Washington, DC, Atlanta, Sunnyvale, and Seattle).

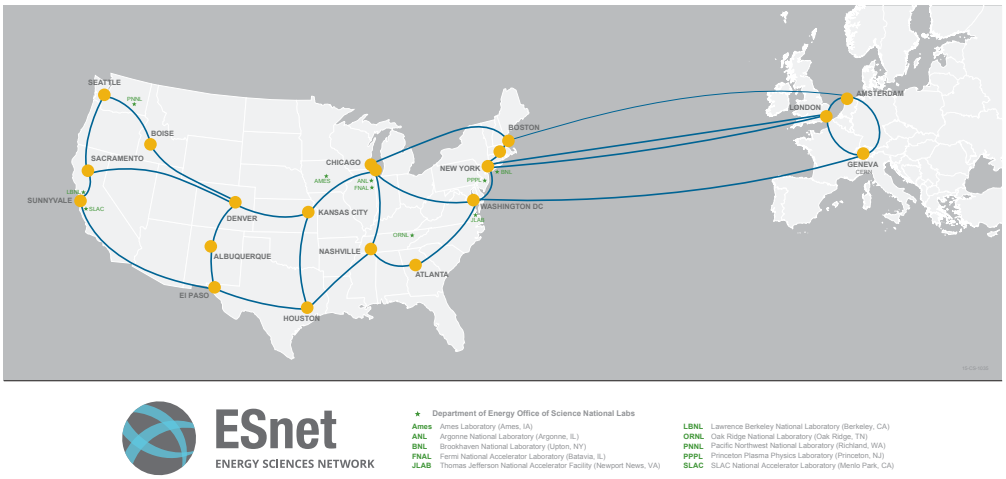


Figure 6.1: ESnet backbone network connects the U.S. science complex nationally and internationally. ESnet's 100Gbps, U.S.-wide infrastructure extends into Europe at 340Gbps to serve the U.S. high energy physics community and other science communities who need high-bandwidth, reliable connections to the science facilities, instruments and data repositories abroad.

Wide-area and local-area network coupling
A second required capability is that of an e ective coupling between the facility's local-area network (LAN) and the WAN to allow for high-speed data transfer. An e ective coupling is virtually never present in a traditionally architected campus network. To overcome this and enable high-speed data movement requires unimpeded access to the WAN network from the systems that have the data. This, in turn, requires a suitable network architecture, routers and switches capable of driving WAN connections at high speeds, and data systems that can moved the data out through a network interface at the required speeds.

A number of DOE national laboratories and universities utilize the Science DMZ [92] concept to address the coupling between WAN and LAN connectivity for large-scale research data. The Science DMZ is a local-area, or institution-based, network architecture that places systems, such as servers, close to the boundary of a LAN and WAN. These servers, also called data transfer nodes (DTNs), are specifically configured to run scalable, high-speed data transfer software (i.e., GridFTP, Globus or bbcp) and are secured through access control lists (as opposed to firewalls) to obtain the necessary data transfer performance for data ranging from hundreds of gigabytes to petabytes. Storage for the data transfer nodes are either locally found on the server (multiple hard drives) or the servers are mounted to a massive parallel file system.

The Science DMZ model increases data transfer performance by simplifying the infrastructure used to support the data movement, using the appropriate data transfer tools on dedicated systems, and employing appropriate security technologies which provide security without compromising performance.[1]

Data transfer tools
The servers, or DTNs, in the Science DMZ environment must use appropriate tools for high data through-

[1] http://fasterdata.es.net/science-dmz/science-dmz-security/.

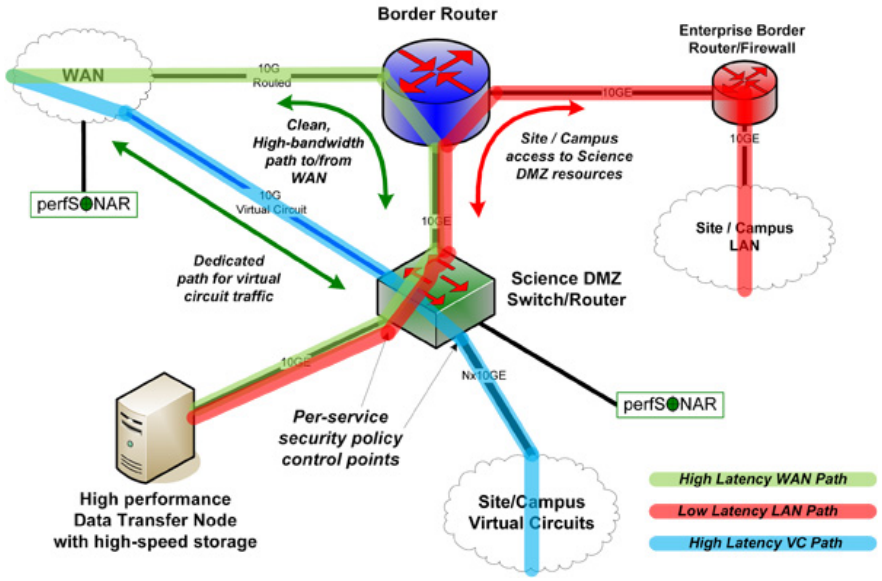


Figure 6.2: A basic Science DMZ architecture. Data travels from the wide-area network (shown in green) through the border router and through a Science DMZ switch which implements the appropriate security policies for the sites.Attached to the Science DMZ switch or router is a server tuned for wide-area data transfers (usually running a set of transfer tools like GridFTP, Globus, bbcp, FDT, etc). Storage for the transferred data is either located on the server or mounted to a parallel filesystem (such as GPFS or Lustre).

put.[2]

DTNs are system implementations that have dedicated hardware and software that are tuned for high-data throughput. For example, a well-tuned data transfer node can achieve transfer rates up to 39.5 Gbps for a memory-to-memory transfer with 4 x 10 Gigabit Ethernet network interface cards, and 9.2 Gbps (1.2 GB/s) for disk-to-disk transfers.

Another example of a state-of-the-art, high-throughput data transfer tool is Caltech's FDT (Fast Data Transfer). FDT manages a collection of attached storage devices (disks, solid-state drives, etc.) through independent CPU threads and multiple, parallel TCP streams. This approach has demonstrated international distance data transfers at 60 Gbps [93, 94]. FDT has been integrated with the LHC CMS data movement tool, PhEDEx.

In summary, high-speed, high-volume data movement is now achievable over distances of 10,000 km and more if the systems, sites, and networks are correctly designed and managed.

Science community-centric networking
Science communities that make use of highly distributed data analysis and management approaches, such as the LHC, have found that a semi-private network environment has considerable advantages. Such semi-private networks allow for more uniform security policies, which, in turn, make it easier to implement and operate Science-DMZ-like approaches to high-speed data transfer.

The LHCONE[3] (LHC Open Network Environment) private network has evolved over several years to provide such an environment to the most of the LHC Tier 1 data centers and large Tier 2 analysis centers. These centers are in operation at scientific institutions across Europe, North and South America, and Asia.

[2]Several data transfer tools are described at http://fasterdata.es.net/data-transfer-tools/.

[3]LHCONE was developed by an international consortium of research and education networks. ESnet has played a leadership role in this collaboration and providing the L3VPN services in support of the LHCONE. More information is available at `http://lhcone.net`.

The approach uses a Layer-3 (L3) VPN (virtual private network) that is an overlay on the general research and education (R&E) networks. The L3 VPN is implemented using Virtual Routing and Forwarding (VRF). The VRFs are private routing instances that are set up in the R&E provider networks (e.g., ESnet, Internet2, etc). The VRFs manage routing for a highly restricted part of the Internet address space that is only accessible to the participating institutions.

In the global LHC community this has proved quite successful at easing the operation of the distributed data centers. The Belle-II experiment at the High Energy Accelerator Research Organization (KEK) in Japan, and its data analysis and management community are now also using LHCONE.

Other instances of such overlays are now being used by several other science collaborations, and still others are considering such an approach.

Challenges

- Current data rates are insufficient for planned facilities. For example, SLAC's current 100Gbps link to NERSC via ESnet is used to offload part of the LCLS science data processing. Current LCLS data acquisition rates are up to 50Gbps. One hundred gigabits-per-second to NERSC will not be enough for LCLS-II, and terabit capabilities will be required if LCLS relies on NERSC for processing LCLS data.

  In addition, although the overall average capacity of the wide-area network may be high compared to the projected usage, this does not address potential "hot spots"—network congestion around high-use facilities like NERSC which can develop unpredictabilities based on disparate experimental facility usage.

- Fast data transfers need to be improved between various scientific users and the scientific user facilities—especially the HPC centers like ALCF, NERSC and OLCF.

- Also for smaller, more diverse scientific facilities, computing resources need to process large data sets or need ways to transfer large data to institutional computing resources. For example, increasingly, with larger radar data sets and especially new high-resolution modeling output, users will need mechanisms to transfer large amounts of data to the computing resources where they can do analysis. 5–10 PB/year in a few large data streams is expected.

- There are communities with considerable growing needs that may have not been factored in to current planning. For example, the Community Earth System Model (CESM) ACME will need to distribute data to collaborators around the world. The current projection shows their distributed data archive reaching multi-petabytes by or before 2020. In the past ESnet has observed that the WAN network traffic generated by a community is directly proportional to the size of the distributed data archive. Based on this we can expect climate to be generating around 100Gbps network traffic in the next few years. Given that the data is used in simulations that are run at the supercomputer centers, this implies that a substantial increase in traffic local to the supercomputer centers could occur in the noted timeframe. This must be taken into account in planning the supercomputer center connectivity to avoid congestion.

R&D needed

- Manual and auto-tuning settings are available but more specific tuning is needed so that the system is dynamically responsive to di erent end pairs at di erent locations.

- Storage system performance is also a significant bottleneck when trying to increase the end-to-end performance for data transfers and science collaborators.

- The LCLS requirement of moving 1 Tbps from SLAC to NERSC in the 2020 timeframe appears to be consistent with the anticipated advances in wide-area network equipment, and, with careful attention, probably also with systems capable of moving such a network stream to disk at NERSC.

– This is likely a case where a Science DMZ approach of some sort is needed. Consulting with ESnet engineers can identify any issues not currently addressed by the standard architecture, and work to update the architecture to reflect these.

Research Challenges: 7

Scientific Data Management: Workflow

Scientific workflows have become a cornerstone for scientific discoveries since it captures the flow of computation and related data dependencies. The term workflow is used broadly in the context of experimental and observational data to refer to a) a complete cycle of scientific discovery, b) the process flows that execute over distributed resources, and/or c) the computational pipeline that runs on HPC systems. These workflows in the EOD system results in complex data processing pathways that are managed today with *ad-hoc* tools and scripts. Increased data volumes and rates, the dynamic nature of the data-driven workflows, the complexity of the processing pipelines in EOD coupled with the increasing complexity of the computational and networking landscape poses serious concerns for the EOS community. The EOS community is at the risk of not being able manage and process the large volumes of data that are being generated at the experimental and observational facilities. *There is a critical need for new technologies that enables automation of the dynamic, real-time data-driven pipelines, supports data sharing and facilitates knowledge transfer of the processing pipelines while ensuring performance, resilience and reproducibility.*

The EOD workshop identified many short-term and long-term needs in workflows that need to be addressed by future R&D. The workflow discussion in this workshop was focused specifically on EOD. A previous workshop, *"The Future of Scientific Workflows"* [95], looked at broader scope of workflows from both the simulation as well as experimental side. There was a diversity in the workflows represented at the workshop; however, common patterns were discernible in the requirements that need to be addressed with additional research and development activities. Three major themes emerged from the discussions at the workshop. First, workflows provide a powerful construct that can be used as a vehicle of knowledge transfer of data and process that is extremely important in EOD pipelines. Second, while there is a diversity of needs there are a few common patterns that can be identified across workflows. Additionally, workflows provide a vehicle to provide the optimization and steering needed in the EOD and HPC environments for qualitative metrics such as learnability, usability, manageability and transparency as well as for quantitative metrics for performance, reliability and scalability. Workflows need to work seamlessly for user needs that requires us to solve a number of research challenges at the boundaries of workflows with data, resource, operating system and programming environments. In this chapter, we discuss the three themes in the context of current state of art, challenges and R&D needed in greater detail.

7.1 Identifying Usage Commonalities and Workflow "Execution" Patterns

There are many di erent types of workflows and usage models in the EOD ecosystem. The workflows span di erent functionality including data collection, assimilation, sharing and comparison with modeling and simulation. Previous work considers workflow execution patterns based on the control-flow of execution and size of workflows [96, 97]. Early patterns of commonalities emerged in the discussion at the workshop. However, a deep-down study of commonalities and di erences across the EOD workflows will be necessary to identify future R&D areas. We need to consider dimensions of time, size and context of use to identify workflow and execution patterns. The patterns will in turn provide hints and impact workflow automation, metadata and provenance. For example, *in-situ* analysis might require more automation but the data collected might not have significant metadata to be associated with it. On the other end of the spectrum, archived data workflows might need lesser automation since it requires human intervention and more metadata which would make it useful for access in the future (§20).

State of the art
Workflow tools have been developed to represent and run scientific processes in a distributed grid and HPC environments (e.g., see research by Kepler [98, 99], Taverna [100], Pegasus [101], Triana [102], Tigres [103, 104]). These tools allow users to compose and interact with workflows, provides seamless access to distributed data, resources and web services. Previous work has provided a taxonomy for scientific workflow systems that classify systems based on design, scheduling, fault tolerance and data movement [105] and characterized a collection of scientific batch-pipelined workloads on processing, memory, and I/O demands, and the sharing characteristics of the workloads [106]. Previous work has also studied characteristics of complex scientific workflows and represented a qualitative classification model to capture the features of a workflow [97].

Challenges
Scientific workflows and workflow tools have been used for over a decade. However, the focus of current technologies has been largely on distributed and HPC simulation workflows and focuses on capturing the computational elements of the scientific process. There is a limited understanding of EOD workflows and the complex relationships between the people, processing, data and resources.

R&D needed
The workshop provided a forum to discuss some of the commonalities and di erences in workflows in EOS. Additional work is needed to understand these workflows in depth through deeper user engagement and user research methods. We identify a number of research topics in this space.

- **Developing a classification for experimental and observational data and process workflows.** We need to build a common understanding of the "data" workflows and understand its context of use and derive commonalities and understand the di erences. The lifecycle of data and the processing of data needs to be captured. EOD projects (§) have a wide variety of workflow needs to manage the data generation, movement, sharing and analyses. User capabilities and needs from the workflow tool in each of these stages is di erent. Many of the projects identified (§21 and §12) that while they had teams that were dedicated to running production workflows, they still had many unmet workflow needs. For example, the teams reported difficulties with running and managing processing and data at scale. Additionally, participants noted the lack of tools for analyses workflows. There is wide variety of needs from supporting developers and support sta to run workflows to end-user interfaces to supporting scientists who need tools to develop and iterate their pipelines. The workflow needs for EOD workflows are more diverse and critical than traditional simulation workflows on HPC systems. Thus, it is unclear if there is a one-size-fits-all solution for all classes of EOD workflows. However, clear patterns of commonalities emerged in the workshop discussions that will serve as the basis for further development of algorithm, methods and infrastructure to support the needs of EOD workflows. There is a need to develop a classification of the needs and understand where existing

workflow tools capabilities can be used and identify gaps where new R&D is needed to support EOD workflows.

- **Develop tools to support optimized scalable and reliable workflow execution patterns in EOS.** Identifying the patterns of execution in EOD workflows provides a unique opportunity to research and develop workflow constructs and execution patterns that capture the needs of the user and meet the required efficiency levels. For example, *ad-hoc* interactive data analyses workflows can be setup to seamlessly execute through queues with lesser wait times. Automated production workflows on the other hand might require higher reliability rates. Today, workflow tools are considered to be complex and attempt to provide a general solution for a wide range of solutions. Workflow classification with supporting tools provides a unique opportunity to create an ecosystem customized to specific user needs that scales to projected data sizes and provides the ability to ensure reliable execution.
- **Identify and develop algorithms, methods and infrastructure that allows similar process and workflow for observation and simulation data.** Once the classes of workflows are identified, additional work is necessary to identify and develop corresponding algorithms, methods and infrastructure. One such class of workflow needs that was apparent across the use cases at the workshop, was the need to allow users to transition between simulation data and experimental data in their processes and workflows. Support for a seamless transition between the two sources of data will require innovation at multiple levels of the software stack including common algorithms, methods and infrastructure (additional details in §2 and §4. Specifically, workflow infrastructure will need additional support to capture and handle the commonalities and di erences in the data, process and resource requirements while providing a seamless view for the end user. Workflows will need to capture and use relevant metadata and provenance that would aid in capturing the di erences and commonalities in the process.

7.2 Workflows to Capture Lifecycle of Process and Data, Facilitating Knowledge Transfer

The scientific use cases identify the need to capture the lifecycle of data and associated provenance across di erent stages and enable users with various skill levels to be able to access and process the data and associated knowledge. Workflows provide a convenient vehicle to both capture and enable knowledge transfer across individuals and groups.

State of the art
Workflows have been used for representing the computational process on HPC and distributed resources [107]. Workflow provenance [108, 109, 110, 111, 112, 113] has been used as a way to capture and track lineage information (more discussion in §9). Domain specific workspaces have been developed to share workflows [114, 115] and communities have developed workflow repositories [116, 117]. However, workflow and knowledge transfer in scientific communities largely happens in *ad-hoc* ways. Previous work has not focused specifically on using workflows to facilitate knowledge transfer.

Challenges
EOS communities recognize the need to capture the lifecycle and associated provenance and facilitate knowledge transfer across various participants of the ecosystem. EOD workflows have specific challenges that need to be addressed in the workflow capture process—workflow representation has to be beyond organizational and facility boundaries and needs to capture the process beyond the computation.

R&D needed
Today, scientific workflows are used to capture the computing processes and data dependencies. During execution, workflows often capture the provenance and metadata associated with execution. However,

as we explore the use of workflows to capture the lifecycle of process and data, and facilitate knowledge transfer, there are a number of research challenges that need to be addressed.

- **Identifying and developing the mechanics to capture the lifecycle for EOD workflows.** There are a number of open research challenges when it comes to using workflows as a knowledge transfer vehicle. First, we need to identify what crucial elements of the end-to-end processing that consists of the people, resource and data that crosses organizational and facility boundaries. It is also necessary to consider other dimensions like time. Next, appropriate constructs to capture these elements need to be determined. For example, are directed acyclic graphs, relational graphs or databases appropriate interfaces to capture the lifecycle of the data and processing? How do these constructs then support data operations (e.g., archiving) or function (e.g., data dissemination). It will also need to be identified how well these workflows can provide to capture provenance and metadata annotation and reputation support for the events in the ecosystem (additional discussion in §9). The constructs need to be validated and optimized for queries resulting from identified use cases for both usability as well as efficiency.
- **Enable sharing across users, organizations and collaborations.** There is a need to go beyond thinking of sharing workflows as a way to share the tools and infrastructure—instead some projects require sharing the knowledge of the process and data. There are research questions that need to be addressed to identify the modalities of sharing across users, organizations and collaborations where workflows can be used as a vehicle of knowledge transfer. How can workflows be used for training and sharing? What needs to be captured in workflows to support sharing of knowledge? Can workflows enable creating community catalog of software libraries and processes for workflow composition and mapping data?
- **Support the goals of experimental and observational collaborations for learnability, usability, manageability and transparency.** As workflows are used for knowledge transfer, they need to meet the goals of the collaborations for learnability, usability, manageability and transparency. Collaborations want users to be able to quickly get to the data and learn about the data and processing and go from quick canned analyses to their own analyses. The workflows that are shared in a collaboration must be easy to use and manage on a variety of platforms and supercomputing centers. Workflows need to provide the right level of transparency—hide the complexities of the infrastructure but allow the user to customize it for its own needs. To enable workflows to meet these goals, research will be needed in workflow technologies with a focus on human computer interaction in context. We will need to develop appropriate interpretations of these metrics and develop techniques to evaluate these user goals are being met.

7.3 Optimizing Performance, Meeting Real-time Goals and Steering Instruments

Workflows provide a way to capture the dependencies between the data and capture the entire lifecycle. Thus, workflows are a convenient vehicle for the optimization and steering of processes to meet both qualitative and quantitative needs. We expand on some of the challenges and R&D opportunities identified in the previous workflows workshop report [95].

State of the art

The amount of data generated at scientific facilities and associated instruments has resulted in the use of HPC centers for their computational requirements. For example, recent improvements in detector resolution and speed have resulted in unprecedented data rates at the Office of Science's Basic Energy Sciences' national light source and neutron source facilities [118]. The data is transferred to HPC centers and processed for both real-time analyses as well as fine-grained analyses. In the Palomar Transient Factory, data

taken with the camera are transferred to automated pipelines at NERSC using ESnet [119].

EOD workflow tools and infrastructure have been developed for specific use cases (§21 and §12). The workflow system provides data access, management and analysis capabilities. The workflows are managed based on the automation needs (e.g., near real-time processing) and user-triggered actions. Typically, they also provide extensive monitoring of the distributed workflows for system operators, resource providers, and end users.

Challenges
EOD workflows need to seamlessly incorporate the scientific instrument, network, storage, and compute resources to provide the scientist real-time access to the data and processing. Data analysis should match the rates at which data is generated. Scientists should be able to inspect results and make near-real-time decisions to modify the analysis or control the instrument. There are a number of challenges in coordinating the resources and data to result in the coordinated efficient execution and dissemination of data. For example, this will require the coordinated reservation of a diverse set of resources such as an instrument end-station, local storage infrastructure, wide-area network bandwidth, remote storage, and remote compute, that is not possible today.

R&D needed
Scientific workflows provide a unique opportunity to meet user needs and system level requirements. Users need seamless access to data and resources to process the data. System usage needs to be optimized for efficiency. There are a number of challenges at the boundaries of workflows, data, resource management and underlying programming models and operating systems that need to be investigated further (additional discussion in §8 and §6). We provide a summary of the topics that were discussed in the context of the workshop.

- **Provide seamless movement between experiment environment and computational environment.** The EOD workflow should incorporate seamlessly the instrument, the HPC center, network allowing the scientist to move back and forth between the experiment and computational environments. While there are a number of technical issues that will need to be addressed at the facilities (e.g., single sign-on, authorization), science use cases identified the workflow as being the vehicle they see as addressing these needs. It will be necessary to investigate how the end-to-end workflow can be used to represent the interaction between the environments, facilitate the human in the loop and track the transitions and related provenance (related discussion in §9).

  EOD workflows often harness resources at one or more computational facilities. The coupled experiment-computation system provides a rich source of opportunities and challenges for workflow technologies that need to be addressed. These experimental workflows focus on organizing, moving, analyzing, sharing, and tracking large quantities of data. Additional research is needed to allow for data-driven processing, the ability to track and search data products, integrate experiment data with other knowledge sources (related discussion in §1).

- **Facilitating data sharing through workflows.** Data is central to EOD workflows. EOD workflows are triggered by the data generated at the instruments and facilities and produce a number of data products during the processing. These workflows face various challenges when it comes to managing a shared data space for their groups or community of users. Various users at the workshop identified the need to have shared storage spaces (akin to file sharing services like Dropbox, Box) that can be used during workflow execution. Previous work [120, 121] has investigated data spaces in the context of *in-situ* and simulation workflows; experimental and observational data has di erent characteristics including data generation and sharing semantics.

- **Develop methods and algorithms to meet the performance goals (e.g., real-time needs), reliability, and scalability of EOD workflows.** EOD workflows have a number of quality-of-service requirements. It is important to investigate predictive performance, resource provisioning, and real-time aware scheduling in the context of workflows in conjunction with the facilities' policies. It is also

important to investigate how real-time resources can be scheduled and instruments are incorporated in the workflows to allow users to interact dynamically and adaptively with their instruments and computation. Resource management systems and schedulers, and HPC systems will need to be re-designed to allow for interactive, dynamic, data and event-driven and real-time workloads to be able to get the quality of service desired.

Research Challenges: 8

Scientific Data Management: Storage

Data volume, velocity and variety are growing across nearly all experimental and observational data science domains. The scientific community must therefore leverage fast, scalable, cost e ective, and flexible storage solutions in order to keep pace. Addressing these challenges will not only require collaborations between the computer science and scientific communities, but a long-term engagement to ensure that computer science solutions continue to be viable over time.

8.1 Findings

EOS projects produce growing volumes of diverse and irreplaceable data. Experimental data must be ingested or bu ered immediately to avoid the loss of key scientific observations. Once the data is ingested, it may have a long (or even indefinite) useful lifetime, which amplifies the need for careful stewardship and indexing.

EOS data is frequently stored and processed on systems that were optimized for other purposes. For example, EOD data is often stored and processed at HPC facilities. This arrangement is cost e ective, but it forces EOD data access and analysis to conform to interfaces, semantics, and quality of service guarantees that have been optimized for entirely di erent workloads such as batch parallel computing applications.

Emerging storage technologies have the potential to improve efficiency. Examples include non-volatile random-access memory (NVRAM) storage devices, cloud storage, and deeper memory and storage hierarchies for large-scale systems. We must lower the barrier for adapting new technologies so that they can be more quickly and broadly used by the community.

Data access latency is an impediment to scientific productivity. Data becomes less valuable if it cannot be written or retrieved in a timely manner. This is particularly true for distributed collaborations in which scientists utilize data captured at a remote facility.

EOS often depends upon the combination of observational data and simulations. We cannot consider these two topics in isolation; many scientific workflows rely upon the sharing of data between experimental sources and computerized simulations.

8.2 Immediate storage requirements and challenges

This section describes the present and near-term barriers to scientific productivity for experimental and observational data management and storage. These issues are closely related to the challenges and R&D needs of service facilities (see §6.2 [1]).

8.2.1 Storage and data processing models

Experimental and observational data is often stored on large-scale parallel file systems at computing or data facilities, but these file systems are not optimized for the ingest or processing of such data. Scientific storage system procurement is instead driven by batch parallel workload requirements and legacy interface compatibility; as a result there are few (if any) mechanisms to di erentiate services or policies for streaming data and analysis workloads. There is an opportunity to make better use of available resources using data models, runtime services, and processing mechanisms that more closely match the needs of the EOD community.

State of the art
While some facilities have developed a storage model that is optimized specifically for analysis of experimental and observational data (for example, the dCache system used by Deutsches Elektronen-Synchrotron (DESY) and Fermi National Accelerator Laboratory (FNAL), Figure 19.2), most facilities leverage parallel file systems that were designed primarily for HPC use. Existing parallel file systems provide a single coherent namespace with uniform semantics, reliability, and interfaces for all users. Experimental and observational data workflows are mapped onto this environment alongside conventional HPC applications.

In the computer science community, it has been demonstrated that big data applications and Internet services are most likely to adopt and productize alternative programming and storage models and semantics (e.g., Spark, Hadoop, S3, Cassandra, Accumulo, etc.) that are more closely tailored to specific use cases.

Challenges
Di erent science use cases could be comprehensively optimized by deploying a specialized storage system for each use case. This deployment strategy is impractical due to the complexity and management cost, however. There is a fundamental tension between the desire to present specialized services and the reality of providing cost-e ective management as illustrated in Figure 8.1.

Experimental and observational science is also characterized by strong quality of service requirements that are not evident in batch HPC or Internet service domains. Data streamed from one-of-a-kind scientific instruments must be stored and processed in a timely manner or else irreplaceable scientific data will be lost.

R&D needed

- Identification of simpler data models (e.g., object storage in conjunction with science-specific indexing) that can meet EOD requirements with lower overhead than conventional file system abstractions.
- Design of pluggable/customizable data models and services that can leverage shared storage resources for streamlined administration yet present di erentiated services and data models for EOD use cases.
- Efficient support for uncoordinated, serial workloads, particularly examples such as sensor data storage that exhibit small data or metadata access patterns. Present-day parallel file systems may com-

[1] Section 6.2 in particular highlights deployment, policy, user access, and efficiency issues that should be considered in tandem with emerging storage technology.

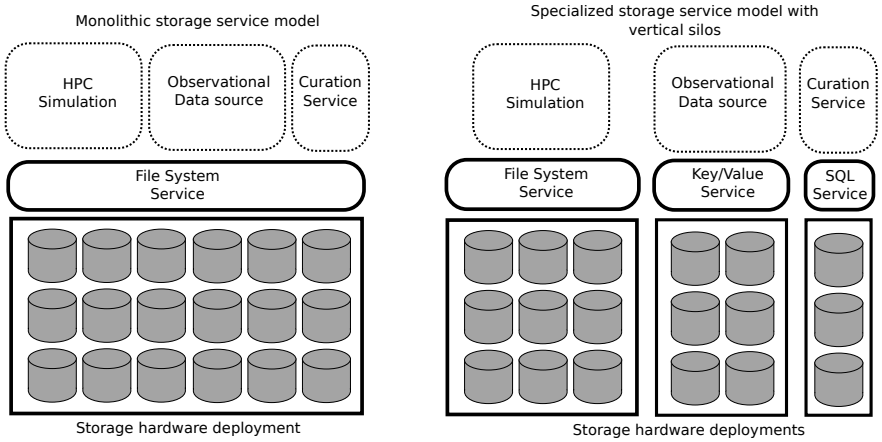


Figure 8.1: Examples of conventional storage hardware and service deployment strategies. In the left figure, all EOD and HPC users share a common storage service. In the right figure, independent storage silos are used to di erentiate data models and service requirements for key use cases. An ideal storage platform would combine the cost and maintenance benefits of the former with the customized service benefits of the latter.

promise or neglect this workload in order to better optimize for coordinated bulk access from parallel applications.

8.2.2 Indexing

Scientific data must be indexed and presented to users in a manner that facilitates query and retrieval. Indexing in this context refers to referencing data according to attributes, scientific characteristics, or experimental parameters such that relevant data can be located quickly via queries based on those properties. A conventional file system imposes a very limited form of indexing and organization, in which relevant parameters must be encoded into the file and directory names for the data. Additional multidimensional indexing can be a very costly component of the scientific workflow.

State of the art
A large volume of experimental and observational data is explicitly loaded from its native format into a dedicated database for indexing, querying, and retrieval.

The HPC community has explored a number of methods for indexing and organizing scientific data produced by simulations. Examples include FastBit [122] by Wu *et al.*, Scientific Data Services [123] by Dong *et al.*, and evaluations of SciDB [124] by Yao *et al.*

Challenges
The process of transforming or loading data into a dedicated database is too costly both in turnaround time and hardware costs for large data sets. The fundamental problem is that it requires creating an additional copy of the data and transforming it into a di erent format. If the database contains references to original files (rather than a copy of the data itself) then there is a danger that the files may become out of sync with the database references, and finding and accessing data will incur the overhead of accessing two distinct storage systems.

R&D needed
Research and development is need to identify additional streamlined and broadly applicable methods for indexing and retrieving scientific data "in place." Ideally, the end user would be presented with an appli-

cation programming interface (API) that resembles a traditional database but provides coherent access to data stored *in situ* in its native format in order to avoid the explicit and costly extract, transform and load (ETL) process.

8.2.3 Access latency

Access latency in this context refers to both the time required for data acquisition from an experimental and observational system as well as the turn around time for scientists to acquire the experimental and observational data that they depend upon for their research. High latency can limit the frequency at which data can be acquired, and is a direct impediment to a scientist's ability to produce timely research results.

This latency may arise due to a variety of factors, including the use of low performance storage devices for capacity or cost reasons, the geographical distribution of large data sets (which may increase the physical distance between scientists and data sets), and the lack of sufficient networking bandwidth at the science facilities.

State of the art
Science facilities utilize a variety of strategies to address the access latency problem. Physical disks and tapes are shipped to and from scientists in some cases (See Sections 18,13). When sufficient wide-area bandwidth is available, ad hoc remote transfers are performed using tools such as FTP, bbcp, and Globus GridFTP (See Sections 13, 18 and 12) or managed (third party) remote transfers are performed using Globus or File Transfer Service (FTS).

Commercial web services and media companies (e.g., Netflix) address latency by designing content delivery networks that take into account usage patterns and the nature of the content to minimize the latency perceived by their end users. For example, frequently accessed data is replicated widely at data centers closest to anticipated users. Similar approaches have been explored by the science community, for example, by the ATLAS Collaboration [125].

Challenges
There are three primary challenges to addressing access latency. The first is simply a lack of network infrastructure for end users outside of major facilities. The second is that in cases where computer science solutions exist (i.e., for wide-area data transfers), they are not well-integrated with the science workflows. Finally, many data centers with sufficient capacity and bandwidth are designed to cater to HPC workloads rather than experimental and observational data.

R&D needed

- Design of flexible storage systems and end-to-end storage architectures that can address the access requirements of both data-centric and compute-centric workloads. In particular, EOD access latency should not be compromised by the presence of bursty HPC I/O workloads.
- Identification of the methods to simplify or generalize the integration of wide-area transfers into automated end-to-end science workflows. Examples such as BigPanDA for the ATLAS project at the LHC and the SPOT Suite for the ALS projects have demonstrated this type of integration for key projects.
- Development of mechanisms to characterize data use and optimize how it is dispersed across available storage resources (both distributed and local) to minimize latency.

8.3 Impact of Near-Term Storage Technology and Disruptions

This section evaluates anticipated challenges and disruptions arising from changes in near-term storage technologies and architectures (e.g., exascale systems).

8.3.1 Emerging storage technologies for data ingest

NVRAM storage devices are a promising technology for use in experimental and observational data ingest due to their low latency and high throughput. They could be used to bu er incoming data from instruments in much the same way that "burst bu ers" [126, 127] will be used in HPC deployments to absorb simulation data before it is transferred to higher capacity primary storage.

State of the art
The first generation of HPC systems with burst bu ers (such as the Cori system at NERSC [128]) are nearing production readiness for HPC workloads. The HPC community and vendors are actively working on the software infrastructure for burst bu ers to make sure that they address the needs of HPC application workflows. Examples include Cray's DataWarp [129] and DataDirect Networks' Infinite Memory Engine [130].

EOD facilities have already recognized the need for similar dedicated storage resources to bu er incoming experimental or observational data and are using a variety of storage systems for these purposes, but these systems do not yet use the NVRAM (or similar) technology (see §15 and §19 for examples in which a dedicated storage bu er is used to capture initial data from instruments).

Challenges
Performance is not the only challenge for EOD data ingestion bu ers. Several of the science use cases call for the ability to inspect and operate on data during its initial capture phase in order to provide rapid feedback to users, quality control, initial analysis, or user-defined triggers on the scientific data. This requires a richer level of functionality than simple transparent bu ering. Additionally, scientific instrument workloads are characterized by continuous streaming more so than the highly concurrent bursts that motivate HPC burst bu er designs.

R&D needed
Development of storage architectures and software solutions that allow emerging storage technologies such as NVRAM to be used during streaming data ingestion while also facilitating in-flight analysis and manipulation. NVRAM devices can simply be procured in place of conventional storage devices to improve performance in cases where ingest and bu ering solutions are already in place, but they exhibit di erent access properties and characteristics. Additional research (for example, to take advantage of lower-latency, byte-granular access modes) will ensure that they are utilized to their full potential.

8.3.2 Mapping data to the storage hierarchy

EOD consists of a variety of di erent types of data (from di erent instruments or experiments, at di erent measurement rates, etc.) with di erent lifecycles and usage patterns. Simultaneously, storage systems are increasingly using a deep storage hierarchy to balance the characteristics and costs of di erent types of storage devices. The convergence of these two trends suggests the EOD workflows will have to e ectively map data (and data processing) to appropriate levels of the storage hierarchy to be able to make efficient use of available resources.

State of the art
EOD data is typically mapped to available data center resources in an ad hoc manner, in response to scientific needs. The computer science community is increasingly exploring deep memory hierarchies from an

architecture perspective but has not yet reached a consensus on a coherent usage model. Specifically, there has been early work exploring the role of deep memory hierarchies for supporting data staging and *in situ* or in-transit application workflows [131, 132].

Challenges

- Each science domain categorizes its data di erently. There is no standardized, uniform way to describe access frequency, QOS requirements, or anticipated access patterns consistently across data sets.
- Application workflows exhibit di erent and varying access patterns and access priorities that may be conflicting, which makes data placement challenging.
- There is no consensus on the cost or utility models (e.g., in terms of performance, energy, etc.) for levels in the large-scale storage system hierarchy.

R&D needed

- Develop a consistent taxonomy of scientific data (how it will be used), achieved either via survey or via automated characterization methods.
- Catalogue data access patterns of EOD workflows and develop mechanisms for identifying these patterns at runtime.
- Develop cost and utility models for multi-tiered storage systems that can be used to optimize data placement and the movement of data across the storage hierarchy
- Develop models in which data analytics systems can be treated as storage endpoints in the mapping process to streamline analytics-oriented data transformations.

8.3.3 Data sharing between experiments, observations and simulations

End-to-end science workflows will require the seamless sharing of data between simulations and experiments or observations as well as with services for analysis, uncertainty quantification, etc. Consequently, storage solutions will have to facilitate such sharing while address the heterogeneity in, for example, data rates, access patterns, execution platforms, etc.

State of the art
The data movement in many application workflows is defined by explicit I/O access to and from a parallel file system. Recent work as part of the International Collaboration Framework for Extreme Scale Experiments (ICEE) project has explored the use of data-staging techniques to support data sharing (and in-transit data processing) between coupled fusion simulations and experiments at the Korea Superconducting Tokamak Advanced Research (KSTAR) facility [133, 134]. From the computer science perspective, frameworks such as DataSpaces [135] can make it easier for applications to implement such end-to-end coupled workflows.

Challenges

- Experiments, observations and simulations may exhibit very di erent data generation and access patterns and the storage solution has to impedance match between them.
- Experiments, observations and simulations are typically located at geographically distributed facilities, and data has to be transported over wide-area networks.
- Data often needs to be transformed at runtime, before it can be exchanged between experiments, observations and simulations.

R&D needed

- Develop distributed storage architectures that can support data sharing between experiments/observations and simulations.
- Develop programming abstractions and runtime systems that can enable data discovery and data sharing between components of end-to-end workflows that integrate observations and simulations.
- Develop of support for in-transit data staging and data processing to address the heterogeneity of data producers and consumers.
- Develop optimizations, such as data prefetching, to reduce impact of data transport and latency.
- Develop efficient data transport mechanisms over wide-area networks.
- Develop methods to facilitate bringing analysis and visualization methods to the storage system (as opposed to transferring the data itself) in order to facilitate collaboration.

8.4 Long-term Data Lifecycle Challenges

This section explores the role of storage systems in data curation: how do we maintain data for long periods of time in a productive and cost-e ective manner?

8.4.1 Facilitating data curation and preservation

Stewardship of experimental and observational data implies not just reliable storage but also a variety of data processing steps such as quality control, generating metadata and provenance information, DOI referencing, indexing, and replica verification. Section 10 also identifies *long-term data preservation* as a critical challenge to preserve the integrity of scientific data over time. Many of these stewardship and preservation activities could be streamlined or automated by incorporating key building blocks within the storage system itself. In this section we focus on storage technologies that can help to facilitate higher level data curation activities.

State of the art
Most data curation steps are performed explicitly in a domain-specific manner. Quality Control (QC), for example, is performed in several locations including at the instrument, during data streaming, or on primary storage. The QC process, particularly on primary storage, often require the data to be explicitly read from storage for processing. Likewise, integrity checks or replica validation require reading data from storage to a compute resource for in-memory verification

Previous work in general-purpose active storage has shown that considerable performance gains can be achieved by performing simple computations or manipulations on data within the storage system itself [136, 137]. Information lifecycle management (ILM) tools in the commercial arena have proven successful in ensuring that invariants or lifecycle rules are automatically maintained for long-lived data [138]. Storage systems such as Amazon Glacier [139] have been used to optimize long-term storage di erently from frequently accessed data to achieve significant cost savings.

Challenges
Current high-performance storage systems do not have any awareness of data curation activities in order to optimize for these activities, possibly di erently than for general purpose I/O. There is also no mechanism to express policies for long-term data storage. Commercial tools and policy engines for ILM are difficult to apply in a parallel or distributed storage environment.

R&D needed

- Further research is needed to identify mechanisms to facilitate curation and preservation activities without transferring data from the storage system to a compute resource.
- Triggers could be associated with the data sets to actively enforce invariant properties on the data. For example, a trigger could be used to indicate that a particular data set should be scanned for integrity or verified against a remote replica on a periodic basis.
- Mechanisms to cross-validate replicas that are stored in di erent formats (e.g., optimized for di erent indexing or analysis methods) would help to preserve data while also supporting multiple data usage models.

8.4.2 Storage federation

The federation of EOD data stores can enable explorations across multiple experimental or observational data sets and broader scientific investigations. Storage federations can also address the growing storage needs in some domains that are producing more data than can be feasibly stored in a single location. See also Section 6.4 which explores resource federation challenges from the service facilities perspective.

State of the art
The federation of data stores is being explored by the computer science community, and several solutions, ranging from loose federations to very tightly and structured federations, have been proposed and adopted by industry. However, their use in science community has been limited to a relatively few large projects. For example the Earth System Grid Federation (ESGF) [140] has a very sophisticated federated storage framework for climate data. Similarly, the LHC experiments have a production federation infrastructure using *xrootd* [125] (and also other approaches using *http*), which o ers transparent access to tens for petabytes of data over hundreds of sites. Tools such as iRODS [141] and CometCloud [142] are being used in some communities to help manage and federate distributed storage resources.

Challenges
Research challenges include scalable mechanisms for flexibly and robustly federating data stores, and for indexing data and/or metadata within the federation so that user can search and discover data across the entire federation, as well as polices for maintaining data integrity, protecting data ownership, and support the required attribution. These challenges are further complicated by the fact that di erent communities can have very di erent requirements, usage modes, practices, as well as policies (e.g., about usage, preservation, data lifetime, etc).

R&D needed
Research is needed to develop appropriate federation protocols and mechanisms that are both scalable and flexible and respect data ownership and access control, as well as "data federation toolkits" providing high-level services that can be used across disciplines to create such federation. Research is also needed in domain-specific metadata schemas and data indexing and querying mechanisms that can enable discovery across a (possibly dynamic) federation. Finally, from a policy perspective, research is needed in data ownership and access policies as well as policies for attribution.

Research Challenges: 9

Scientific Data Management: Metadata and Provenance

In this section, we provide a summary of the discussions surrounding the topics of metadata and provenance. This section references broad definitions for metadata and provenance following [143, ch. 12] [144, § 4.1.6]. *Metadata* is the information about data [145]. It provides the contextual information, description and characterization about data, and can make finding and working with particular instances of data easier. *Provenance* traditionally describes the parentage of a data object, including information, such as source data it is derived from and the procedure that created it [146, 147]. More recently it has also been extended to describe scientific processes in more detail [148].

Metadata and provenance are critical to most aspects of data management including storage, retrieval, analysis, curation, publication, and preservation. At the workshop, the participants representing DOE's Basic Energy Sciences (BES), Biological and Environmental Research (BER) and High Energy Physics (HEP) communities are unanimous in their opinion that metadata and provenance are essential for the analyses of experimental and observational data and the validation of the scientific insights gained. However, today the capture of these critical information largely still relies on manual, non-digital and non-sharable approaches, hindering scientific discovery in increasingly high-velocity, high-volume data environments. Workshop participants identified four broad categories of challenges:

1. Efficient automated capturing of metadata and provenance,
2. Event tagging and real-time analysis of massive experimental and observational data,
3. Scalable accesses for distributed collaborative analyses, and
4. Reproducing and validating scientific outcomes.

Next, we briefly summarize the state of art the key challenges and the R&D needed in each category in turn.

9.1 Efficient Automated Capturing of Metadata and Provenance

Experimental and observational sciences have a centuries-old tradition in capturing scientific processes, their decisions taken throughout the process and its results. Handwritten lab notebooks are the primary capture mechanisms for such information. They provide a crucial basis for result analysis, evaluation, validation and reproducibility. In the last two decades, electronic notebooks have become more prevalent,

however they still require the manual entering of information in a non-standardized format. Some workflow systems could automatically capture certain metadata and provenance of scientific processes, however, such systems could make it easier to share the analysis workflows and results. For example, the UK's LabTrove project has produced a workflow sharing platform (MyExperiment) with an electronic lab notebook that captures the workflow provenance during the analysis [149]. However, even the most advanced systems available today have not been designed to work in extreme-scale data environments.

Metadata best practices
Among the scientific domains represented at the workshop, there are a number of communities with well-organized metadata, however there are also a number of application scientists describing their situations by alluding to the Tower of Babel. Broadly, the communities with large shared experimental facilities or large shared data collections have thought about their metadata practices, while other communities have not established a common vocabulary, process, or standard. For example, the high-energy physics community, climate research community, geographical data community, and cosmology community have well-organized community-wide data processing e orts. The high energy physics community uses a shared data processing platform based on ROOT[1] [150], which prompts a common standard for data and metadata. Their workflows are captured typically as ROOT scripts. The geographical data and earth sciences communities have developed a number of metadata standards including ISO 19115 and ISO 19139. In the United States, there is coordinating organization known as the Federal Geographic Data Committee (FGDC) that is responsible for developing, using, sharing, and disseminating geospatial data through the National Spatial Data Infrastructure (NSDI).[2] Under the auspices of the World Wide Web Consortium (W3C), there is a set of metadata standards under the title of Dublin Core.[3] Though it is more commonly used by the library sciences and other social sciences, some natural sciences communities are adapting aspects of these standards, for example, National Aeronautics and Space Administration (NASA) is actively promoting the use of Directory Interchange Format.[4]

During the workshop, Dean Williams, the PI of the Program for Climate Model Diagnosis and Intercomparison (PCMDI) project[5] and the Ultrascale Visualization Climate Data Analysis Tool (UV-CDAT) project,[6] presented a compelling set of use cases from the climate community. Next, we briefly summarizes these examples to illustrate how metadata might be used and evolved in an actual application.

The climate community has a community-wide standard on metadata, known as the Climate and Forecast (CF) Metadata Conventions that is used in a large variety of data sets [151]. This standard was started by the Atmospheric Model Intercomparison Project (AMIP)[7] project in the late 1980s, and propagated to climate simulation groups, re-analysis groups, and eventually adopted by the wider community. This common standard facilitates the exchange of data from di erent sources, comparison of software tools, and validation of research findings. This standard is an integral part of the Climate Model Intercomparison Project (CMIP),[8] and is key to the Inter-governmental Panel on Climate Change (IPCC) Assessment Reports.[9] In the CMIP project, metadata can appear in three forms: (1) as header information in a netCDF file, (2) as descriptions of data sets entered by users through a questionnaire, (3) as free-form descriptions of the data file or data sets. Data sets entered to CMIP repository also go through extensive quantity control (curation) steps including metadata format conformance checking and file format consistency checking. CMIP software system allows users to report quality problems about data sets and relay reports back to the submitters of the data. After a certain period of error-free uses, a data set is issued a DOI so that users can

[1] http://root.cern.ch
[2] https://www.fgdc.gov/nsdi/.
[3] http://dublincore.org/.
[4] http://gcmd.gsfc.nasa.gov/add/difguide/.
[5] http://www-pcmdi.llnl.gov/.
[6] http://uv-cdat.llnl.gov/.
[7] http://www-pcmdi.llnl.gov/projects/amip/NEWS/overview.php.
[8] http://www-pcmdi.llnl.gov/ipcc/about_ipcc.php.
[9] https://www.ipcc.ch/publications_and_data/publications_and_data_reports.shtml.

refer to the data set in publications. This permanent reference to a data set acknowledges the contribution of the submitters and encourages data reuses and data preservation.

A key function of metadata is to allow users to locate data relevant to their work. To this end, a best practice is to augment the metadata records with some indexing techniques to accelerate the searching operations. For example, in the current CMIP implementation, three di erent types indexing techniques are used: PostgreSQL [152] for high-level statistics, SOLR [153] index for user-facing operations, and THREDDS [154] for subsetting of data.

At this point, the well-organized metadata process in the climate community is largely limited to simulation data. There is a larger variety of experimental and observational data that are managed by many disparate research groups collecting the data. These disjoint experimental and observational data sets are not easily accessible. Researchers from other communities mentioned similar challenges that will be described later.

State of art in provenance capturing The research community on provenance has developed a range of provenance models including domain-specific solutions such as VisTrails [155] in UV-CDAT [156] and generic solutions such as D-PROV [157]. To enable greater interoperability between provenance models a working group for the W3C defined the core specification for an Open Provenance Model (OPM) [158] in 2011. This was followed in 2013 by the release of a second W3C standard Prov-O [159]. It provides a set of classes, properties, and restrictions that can be used to represent and interchange provenance information generated in di erent systems and under di erent contexts. It can also be specialized to create new classes and properties to model provenance information for di erent applications and domains [160]. Some of the communities represented at the workshop, such as the climate community, have since developed a number of OPM and Prov-O ontologies for specific application domains [161]. Others, such as those from the HEP and BES communities are not aware of any widely adopted community standards on provenance. While much research has focused on provenance capture and storage methods, workflow management system developers have been at the forefront of integrating automated provenance capture into their software [162]. Science communities on the other hand often have a long tradition of capturing the provenance of their work in non-digital means, such as Lab Notebooks mentioned a number of times by BES scientists at the workshop. HEP projects vigorously review and test research results as a team before publication, requiring scientists to capture the provenance of their work in electronic and non-digital media to prepare their work for this process. In climate modeling, teams capture their model experiment scripts to enable them to retrace their steps if required.

The methods for provenance collection generally fall into three categories: workflow event listener, application logs, or direct calls to a provenance vocabulary-based API. Workflow listeners provide a means to directly collect and record workflow events, such as workflow identifier start/stop date time stamp, parameters or data used and so on. Workflow provenance is typically asynchronously collected, and workflow events ordered by the calling order, making transitive closure possible. For Application Provenance Vocabulary APIs, provenance is collected through API calls at application execution time, inferring that any provenance collected is reliant on the developer making calls to the API [163, 164]. For log files, event history is derived from either log files or streaming logs at runtime and reconstructed as provenance using a provenance vocabulary API [165]. Logger APIs support logging at di erent granularities: fatal, error, warn, info, debug, or trace. Interpretations of the log file entries is done through a monitor application that analyses the log files and creates the provenance entries [166, 167]. The majority of the available solutions has been focused on the collection of relatively low-velocity and low-volume provenance data. Only a few projects have started to explore high-volume, high-velocity capture mechanisms. One approach uses messaging services such as Apache Axis2 and RabbitMQ [168] or Apache Kafka and AVRO [169] to facilitate high velocity provenance transmission. In distributed or extreme-scale computing environments provenance capture can at times be unreliable [170], leading to incomplete provenance records.

Challenges in efficient capturing of metadata and provenance
E ective metadata capture was mentioned by all application representatives as a critical need. The meta-

data captured here generally include information such as the time a data record is produced, the user who produced the data records, the program that produced the data and so on. Following are a number of specific issues discussed.

1. **Efficiently capture metadata about data movement, correlation, approaches and architectures of multi-source, heterogeneous metadata in a distributed computing environment.** Existing workflow provenance capture solutions are e ective for low-volume capture requirements, however if the community wants to support new provenance applications that require high-velocity provenance capture new approaches, in particular for extreme-scale systems with deep memory hierarchies.
2. **Describing *in situ* data reduction.** With the increasing costs of data movement and the limitations of I/O systems on the road to exascale, workflows for computational science must shift from saving data for *post-hoc* analysis to incorporating various forms of data analysis and visualization while running a simulation, with comparatively little data saved for *post-hoc* analysis. A key challenge hereby is the development of sufficiently descriptive and detailed provenance models to capture adaptive data reduction processes at runtime to enable the further processing of the data, as well as its validation and interpretation *post hoc*.
3. **Both structure and content of provenance records can be incomplete [171], in particular if captured in extreme-scale environments.** Dropped messages can result in missing nodes or edges in the provenance graphs. Additionally, soft, hard or silent errors can lead to missing or incorrect content in provenance messages. Furthermore interrupted or failed workflow processes due to system errors can lead to incomplete provenance graphs.
4. **Lightweight, customizable approaches to capture of metadata and provenance in highly variable, potentially *ad-hoc* processes.**
5. **Capturing the performance metadata could be very useful in providing feedback about the progress of the application and in debugging, optimizing and developing analysis procedures.** An e ective metadata capturing system should have the option to enable capturing of performance metadata.
6. **Enabling users to contribute data and metadata.** In discovery science, the direction and methods applied during data analyses are often driven by events that are discovered in the data or by the intuition and expertise of the scientist(s) that conduct the analysis. However there are to date no provenance capture methods that work in more *ad-hoc* situations and at scale. In particular the capture of human decision points and reasoning are neglected in today's automated solutions.

R&D needed
The automated capture of metadata and provenance remains an open challenge in all three communities, in particular in distributed environments, against the background of high-data volumes and velocity and during *ad hoc* experimental or analytical processes. Research is needed to not only capture metadata and provenance efficiently, but also to ensure the metadata is complete for understanding the data products. This can be challenging because some of the analysis steps are performed *in situ* and could not be easily reproduced—i.e., rerunning a large simulation program. In addition to automatic metadata and provenance capturing, the metadata and provenance management systems should be flexible enough to capture input from human experts in the form of addition, annotation or correction.

9.2 Event Tagging and Real-time Analysis of Massive Experimental and Observational Data

The metadata captured in the above discussion is available to be recorded without significant computing e ort. In contrast, we use the term "event tagging" to refer to the metadata that requires some e ort to generate. For example, on a timestep of a global climate simulation data, the metadata such as the user

who ran the simulation code and the simulation parameters require no e ort to extract, while whether the data contains a tropical cyclotron would require a detection procedure that requires a certain amount of computing e ort [172]. We refer to the identification of tropical cyclotrons as tagging events. Similarly, in an observation, a data record might be an image about a material sample under an X-ray and a feature might span multiple of such images, in which case, a tag may be generated to refer to all these images. Another common term used to describe such an event is a *feature*. Di erent science communities appear to prefer di erent terms, for example, *dynamic metadata* is used by a number of workshop participants. These dynamic features may be extracted after the data is generated, however, as simulations and experiments generate data faster and faster, there is a trend to extract such features while the raw data is first generated or collected before the raw data is written to the relatively slow permanent storage. In fact, in some cases, the raw data might never be written to the slower storage [144].

State of the art in tagging
Given a large set of raw data, tagging the data to identify those data records with special features is a common practice to help scientists to identify "interesting" data records. This practice is heavily utilized in HEP [173, 174] and the biological sciences [175, 176, 177, 178, 179]. There are di erent ways to categorize the tagging techniques, for example, automated vs. manual, and structured vs. unstructured. Typically, the tagging performed by HEP applications are to classify events based on community-defined ontology [173, 174], while many of the genomic applications utilizes both controlled vocabularies as well as free-form annotations. Often, a large user community could contribute annotations to a centralized data collection, such as the GenBank.[10] Additionally, there are techniques to extract information from existing annotations to generate new information [176, 180, 181].

In some applications, the automated tagging is closely associated with real-time data analysis, where the tags are used to make decisions about whether and when to terminate the experiment or observation, or how to adjust the experimental setting for the next run or next round of data collection. Some of the use cases discussed are in the process of automating all or part of the experiment or data analysis [75, 76]. These automations are largely based on the automated tagging or classification of experimental measurements or observations.

Challenges in tagging
Existing systems for producing event tags are all custom-developed software created with extensive programming e orts. Reducing the cost to develop these tagging systems would make it easier for more experiments and observations to automate more steps of the experiment or data analysis.

Often the tags or annotations are collected into central databases, but need to be accessed by a large number of users distributed around the world. For example, the LHC experiments have thousands of users around the world and similarly, global climate research has many thousands of users sharing a large set of climate simulation results. Providing efficient support for thousands of concurrent accesses to these databases is a challenge. The existing querying and indexing techniques may not provide sufficient performance for the real-time needs mentioned above.

In addition, application scientists have indicated that event tagging should be allowed both at facilities where the data records are generated and by user community that performs the analysis tasks. Given that some large collaborations have many thousands of users, e ective support for community input is a challenge.

In experimental measurements, it is important to capture the uncertainty in the measured values. Correctly capturing and propagating the uncertainty information in the analysis process is another challenge. The propagation of the uncertainty in analyzing both experimental data and simulation data could be regarded as another form of event tagging.

R&D needed

[10] http://www.ncbi.nlm.nih.gov/genbank/.

With growing data volumes, the fast and efficient identification of valuable events for further analysis is becoming increasingly important. The growing desire for larger communities to explore the same data, means that this is not only an initial identification challenge, but a management challenge too. Enabling communities to e ectively describe and discover the events relevant to their research in extreme scale feature spaces.

Uncertainty quantification in observational and experimental data, as well as data products derived from these results is rapidly gaining in importance across the EOD communities. However, to date there are no standardized means to capture, express and compare these insights as part of the data's metadata or provenance records.

9.3 Scalable Accesses for Distributed Collaborative Analyses

When discussing the capture of metadata and provenance in Section 9.1, we have touched on a number of issues related to the scalability of capturing and accessing such data. Here we will not repeat the discussion of the state of the art on scalability of metadata and provenance, but instead concentrate on the challenges.

Challenges to scale-up and scale-out
The metadata and provenance standards and practices must be scalable in a number of di erent dimensions. Here are a few examples.

1. The number of cores in a high-performance computer is quickly growing. The mechanisms for collecting data and metadata must be able to accommodate this growth.
2. User analysis jobs often have many steps; propagating the metadata and event tags through the analysis steps can be a challenging issue.
3. When preserving data records for the long term, it would be highly desirable to have metadata and provenance be permanently associated with the relevant data sets for an extended period of time, say, for several decades. How to store them together and what to store in a durable way are challenging questions.
4. As time progresses, new experimental protocols and new measurement devices are developed, the ontologies and procedures for capturing metadata must be able to evolve with these changes. For example, in the astronomy community, it would be highly desirable to have a metadata framework and tools to be reused across di erent sky surveys.
5. The metadata models should be reusable across di erent specialties of a scientific domain. For example, it is highly desirable to have the same metadata models used on both simulation and experimental observations.
6. As more data become integral part of the public decision making process, some critical data sets might be of interest to a large number of people. The relevant metadata infrastructure should be able to scale as the number of users increases.

R&D needed
Research is needed to scale metadata and provenance in a number of di erent ways. The simple version is to scale the metadata capture to a large number of data sources and a large number of data processing pipelines. A more complex version would include coordinating metadata from di erent but related projects. Another aspect of scalability is to allow ontologies, metadata and provenance to evolve gracefully over time. On projects of national importance, it is important to keep a coherent set of metadata for decades or longer. Therefore, scaling the metadata and provenance across time is yet another research topic.

9.4 Reproducing and Validating Scientific Outcomes

The most important use of metadata and provenance is to facilitate reproducing and validating scientific research results. A workflow may be rerun with a set of modified parameters to explore alternative options in planning, auditing and training. Exactly reproducing a data analysis is necessary to validate scientific outcomes and maintain scientific integrity. Next, we briefly discuss the metadata-related challenges in supporting reproducibility.

State of the art
Reproducibility is defined as: "the ability to recompute data analytic results given an observed data set and knowledge of the data analysis" [182]. Metadata and provenance play a key role, as they enable scientists to compare, contrast and validate research. While reproducibility does not guarantee the correctness of the approach or results, it is an essential foundation for validation. As such, this paradigm is being increasingly embraced by publishers [183] and funding agencies.

To easily reproduce a complex scientific workflow, we would like the workflow to be well documented and could be conveniently rerun. However, as mentioned in the previous sections, metadata and provenance are often in the form of lab notebooks, handwritten during experiments and observations. These handwritten notes typically do not rigorously follow any guiding format that would guarantee the information is complete, accessible, or error-free. Electronic Lab Notebooks have made it easier to share and search such notes on discovery processes, but a lack of standard formats makes it still difficult if not impossible to compare, contrast and correlate these notes across di erent experiments. The ability to formally and automatically check any processes is also not available for this form of provenance for reproducibility.

In computer science, a current research area is focused on the reproducibility of numerical results for single applications and the reproducibility of experiments for workflows through provenance capture [155, 184, 185, 186, 187]. For example, part of the DOE BER ACME project's initial investigations are underway to address the reproducibility of experiments and execution, with an assumption that the core simulation codes are numerically reproducible.

Challenges in Supporting Reproducible Research
One key reason for capturing provenance is to make it easier to reproduce a data analysis procedure. However, using provenance to enhance reproducibility is still in its infancy. In communities that use a shared analysis environment, such as scientists from the HEP community typically use ROOT [150] to run their analyses, reproducing an analysis could be as simple as rerunning the same analysis script on the same data files. However, the existing provenance capturing systems often fail to capture all necessary information to rerun the scripts.

When considering the reproducibility for computational applications as part of the scientific discovery process, additional challenges need to be considered:

- Numerical reproducibility [188, 189]: has the algorithm been numerically designed to create the same results if replicated?
- Experiment reproducibility [190, 191]: do we have all the information about the simulation to repeat it?
- Execution reproducibility: can we recreate the execution environment, execution conditions (including system events) and system architectures?

Reproducibility for workflows adds a set of additional challenges:

- Error propagation reproducibility: can we replicate the processes in which numerical- or system-induced errors or di erences are propagated through the workflow tasks?
- Decision reproducibility: can we reproduce decisions made by the workflow management systems and users during the workflow execution that critically influence the outcome?

- Complexity: can we deliver reproducibility at an a ordable cost across all the above the challenges covering a wide range of algorithms, execution environments, system architectures and events?

Additionally, the workshop participants also discussed the following general reproducibility issues:

- Provenance capture framework often does not capture all the necessary information. For example, one might capture the name and version of the program that generated the data, but neglected to capture the information about the compiler, runtime libraries or the OS. Such low-level information often has an unexpected impact on the output produced, and is therefore important to capture. The challenge is to capture such information efficiently enough to allow users to continue their analysis work without noticing the interferences.
- Investigation of provenance models for reproducibility.
- Capturing the uncertainty information in the provenance is important to the interpretation of the results that di er from each other in di erent runs. However, how to express the uncertainty and express the assessment of the results are new research topics.

R&D needed

The reproducibility of scientific research as part of the validation process for novel findings is foundational to all scientific work, today this is a predominantly manual process that does not scale to large data volumes and complex scientific processes. Sharing of metadata and provenance information with a wider community is necessary for reproducible science but difficult. Research on new provenance will be required that is suitable for reproducing experiments, observation and their analysis. This work will need to consider the tradeo s between the required expressivity and detail to serve the intended purpose versus the volume of information produced that could negatively impact analysis performance or the accuracy of the capture. Furthermore, we need to investigate how we can move from a provenance model with one stream of provenance (i.e., from a workflow) to a system where we have many streams that align and intersect at certain time intervals—for example, if we consider capturing information from a complex experimental instrument or workflow. Research will be needed in enacting provenance as part of the validation or reproduction process.

Research Challenges: 10

Data Curation

Digital curation is a proactive process, where data formats, representations and description are continually reviewed and when needed updated to keep them relevant and useful for a set of well-defined designated user communities. Digital curation involves maintaining, preserving and adding value to digital research data throughout its lifecycle. The active management of research data reduces threats to their long-term research value and mitigates the risk of digital obsolescence. Meanwhile, curated data in trusted digital repositories may be shared among the wider research community. As well as reducing duplication of e ort in research data creation, curation enhances the long-term value of existing data by making it available for further high-quality research [192].

Few data centers provide that level of support, the DOE BER ARM program's data management team is one of the centers that provides active curation. The active curation process involves many di erent processes, many of which remain manual to date, others are embedded in the day-to-day operation. More commonly centers or publishers provide data preservation services, were the data is preserved in its original state for an agreed period of time. Increasingly these data objects are assigned DOI's issued through the international DataCite federation and its partners such as the Office of Scientific and Technical Information (OSTI) for DOE Laboratories. The actual data preservation is however carried out by individual institutions, which need to maintain the data. Data preservation requires less e ort, but has the challenge that the community has to maintain the knowledge on how to interpret a specific data format (including the tools that might be required to read, decompress, decrypt, analyze, etc).

In 2013, the Office of Scientific and Technology Policy (OSTP) released a new directive on Increasing Access to the Results of Federally Funded Scientific Research.[1] This directive had the immediate impact that organizations such as DOE started to develop policies and guidelines to further the long-term accessibility of research results. However the directive also stimulated a discussion within the science communities about data curation, in particular what data is worth preserving, how long should it be preserved, metadata, provenance and active curation required to further data reuse, reproducibility of science and verification of scientific discoveries, as well as the costs models associated with long terms data curation. In particular a recent BER Climate and Environmental Sciences Division (CESD) workshop and associated study highlighted the increase of these discussions [193].

During this ASCR EOD workshop participants also identified a clear need for data curation across the DOE SC communities and facilities. However, during discussions, it also became clear that no solution or approach exists within DOE SC that is generally and broadly applicable. The result is many *ad-hoc* approaches, which results in the duplication of e ort, extra expense, and what may be viewed as an unsustainable approach to curation. A focus on sustainable, program-wide approaches for data curation would

[1] https://www.whitehouse.gov/sites/default/files/microsites/ostp/ostp_public_access_memo_2013.pdf.

broadly benefit DOE SC communities in terms of cultivating and sustaining community-centric data repositories, as well as to fulfill legal mandates for making the results of federally-funded scientific research public.

10.1 Findings

Findings from the EOD workshop can be summarized in two main points regarding data curation.

1. There is a clear need for data curation across DOE SC communities, but there does not exist a solution or approach within DOE SC that is generally and broadly applicable. The result is many *ad-hoc* approaches, which results in the duplication of e ort, extra expenses, and what may be viewed as an unsustainable approach to curation.
2. A focus on sustainable, program-wide approaches for data curation would broadly benefit DOE SC communities in terms of cultivating and sustaining community-centric data repositories, as well as to fulfill legal mandates for making the results of federally funded scientific research public.

10.1.1 Keeping large-scale research data collections accessible and reusable through automated and standardized data curation processes

The majority of science communities are only familiar with long-term preservation, but not with the processes required for proactive curation. While guidance on data curation processes exists through centers such as the UK Digital Curation Center, there are no reference implementations or standardized tools available that are capable of providing automated curation services. In particular tools for the curation of extreme-scale data streams have not been researched to date. Today data curation includes a significant amount of manual and human-centric tasks, making them challenging to apply to extreme-scale data environments.

State of the Art
Data curation practices vary significantly between di erent scientific domains, we therefore review in this section state of the art from a number of di erent viewpoints.

The UK Digital Curation Center[2] is the premier source on information and training about data curation. They provide guidelines to assess existing data assets, to plan the setup of a data curation facility and to assess the e ectiveness of an existing facility. The Open Archival Information System (OAIS) is an ISO standard initially developed in 2000 by the space science community, which describes the organization of people, processes and systems required to run a long-term data preservation service.[3] However, while standards and guidelines have been in existence for some time, no reference implementations or standardized tools exist today for digital preservation and curation. The communities that engage in digital curation usually develop their own *ad-hoc* processes and tools to provide the required services to their communities.

BES community
Data curation or preservation is today seen as the task of the individual PI, which the exception of the neutron facilities, that have a tradition of preserving raw data for their community as the data volumes are seen as small enough to make that easily feasible. In the future, there is a requirement to store data and metadata, use universal data formats, and provide physics-based curation (see §17). Data availability and curation would be fundamental in enabling the future reuse of data for scientific verification, integration of experimental results to create meaningful statistical insights, create a knowledge base that can help

[2]UK Digital Curation Center, initially funded as part of the UK eScience Program led by Tony Hey; http://www.dcc.ac.uk/.
[3]Open Archival Information System (2012 update); http://public.ccsds.org/publications/archive/650x0m2.pdf.

experimental planning and real time analysis. In Europe, the Pandata project has started to address some of these issues to enable facilities to establish long-term knowledge archives, not only at a single facility, but also across facilities.[4]

High Energy Physics community
In 2009, the HEP community formed a Data Preservation in HEP (DPHEP) study group that investigated whether the community had a need for a planned approach to data preservation and curation. The study concluded that there would be great benefit to the long term preservation of results, as evidenced by recent discoveries that had been enabled through the reanalysis of data from a prior experiment. The study group then examined what actions would need to be taken and concluded in its 2012 report[5]:

- Urgent action is needed for data preservation in HEP.
- The preservation of the full capacity to do analysis is recommended such that new scientific output is made possible using the archived data.
- The stewardship of the preserved data should be clearly defined and taken in charge by data archivists, a new position to be defined in host laboratories.
- A synergistic action of all stakeholders appears as necessary.
- The activity is best steered by a lightweight organization at an international level.

The group further recommended the following actions:

- Priority 1: Experiment level projects in data preservation.
- Priority 2: International organization DPHEP.
- Priority 3: Common R&D projects to develop preservation models and tools.

Astronomy community
The astronomy community has a long tradition of long-term data preservation for its observational studies, and they rely critically on data sharing and the reuse for its scientific work. The UK Digital Curation Center provides an excellent summary of key data formats, tools and active data archives.[6]

The National Virtual Observatory and its successor, the Astronomical Virtual Observatory (http://usvao.org) set out to federate the data from the whole astronomy community, by defining a set of lightweight protocols that were easy to implement by most data providers. These grew into a world-wide e ort, called the International Virtual Observatory Alliance (IOVA).[7]

Several large projects, like the SDSS, have attracted a wide-range of users, who go considerably beyond the professional astronomy community. An interactive database access was provided to the users who were able to make observations through a virtual telescope which was online day and night. Over 5,000 refereed papers with 200,000 citations resulted in the use of this data set to date. Citizen science projects like GalaxyZoo[8] have attracted hundreds of thousands of internet scientists who participated in the research using the open data, and made several major original discoveries.

Large cosmological simulations are also turned into the Open Numerical Laboratories that can be used interactively, and are becoming the norm for the community. The simulations are presented through an intuitive databases interface that enables the users to query complex evolutionary histories (merger trees) of the galaxies. The first of such databases was the Millennium simulation database, followed by others. These methods provide a novel access pattern that go beyond just downloading the simulation snapshots.

[4] Experimental Science at Large Scale Facilities—European Pandata Open Data Infrastructure; http://pan-data.eu/PaNdataODI.
[5] 2012 Status Report of the DPHEP Study Group: Towards a Global E ort for Sustainable Data Preservation in High Energy Physics; http://arxiv.org/ftp/arxiv/papers/1205/1205.4667.pdf.
[6] http://www.dcc.ac.uk/resources/metadata-standards/disciplinary/astronomy.
[7] IVOA: http://ivoa.org.
[8] http://galaxyzoo.org/.

The users do not even need to know whether the underlying simulations are 10 TB or 10 PB, as long as the results of a query appear rapidly.

Climate and Earth Sciences Communities
This is a community with a long tradition in data preservation and curation in particular for their observational data collections. For example the World Data Centre (WDC) system was created in 1957 by this community to archive and distribute observational data. At the end of 2008, the World Data Centres were reformed and a new International Council of Science (ICSU) World Data System (WDS) was established in 2009 that encompasses a wider set of scientific domains. ICSU WDS promotes universal and equitable access to, and long-term stewardship of, quality-assured scientific data and data services, products, and information and coordinates trusted scientific data services for the provision, use, and preservation of relevant data sets. ICSU utilizes the Committee on Data for Science and Technology (CODATA) to develop strategic collaborations on issues of common interest.

Under the World Climate Research Programme (WCRP), the Working Group on Coupled Modeling (WGCM) established the Coupled Model Intercomparison Project (CMIP) in 1995 as a standard experimental protocol for studying the output of coupled atmosphere-ocean general circulation models (AOGCMs). CMIP provides a community-based infrastructure in support of climate model diagnosis, validation, intercomparison, documentation and data access. The Earth Systems Grid Federation (ESGF), originally developed in the late 1990s, is an international collaboration that provides hereby the underpinning infrastructure for long-terms data preservation, discovery and sharing. Data in the ESGF infrastructure undergoes strict data quality checks and have to adhere to fixed data and metadata standard requirements.

However despite the e ort of these data centers and many others that are not directly connected to the WDS or ESGF collaborations, there are no publications, standardized approaches or tools in this community for long-term data curation and each center continues to develop their own *ad hoc* approaches. The advent of dramatically increasing data volumes and rates presents a significant challenge to the community to continue their existing data curation e orts.

Biology community
The biology community shares its data predominantly through databases, some very large and standardized (e.g., the Protein Data Bank, PDB; Proteomics Identification Database, PRIDE; and Gene Expression Omnibus, GEO), but the majority of the databases are small and usually maintained by single investigators. These databases are often developed and curated over many decades. Traditionally, di erent fields of biology define their own specialized data and metadata in the form of "minimal information" standards, together with the accompanying specialized software. More recently the need for interdisciplinary projects has led to the development of ontologies that help to link across some of these formats for specific topic areas, such as the Gene Ontology. A 2009 talk from one of the world's leading biology institutes[9] describes the predominantly manual data curation e ort in this complex environment. These observations lead to the formation of the European ELIXIR project,[10] that aims to establish a pan-European data infrastructure in support of life sciences research. Since then, there has been an e ort to develop an overarching framework for data curation in biology based on the ISA-Tab standard (Investigation-Study-Assay) together with a suite of software tools to support standard-compliant experimental annotation and community-level data curation[11] The ISA-Tab format is intended to record the minimal information needed to understand how a biology data set was generated, but still depends on other community-based standards to describe the data itself. To facilitate this, a biosharing resource was created as an outgrowth of the ELIXIR project to provide access to the most current community standards (BioSharing). These e orts to standardize biology data to facilitate curation have been adapted by several new data-oriented publications as the preferred format for biology data submission (e.g., Nature Publishing group's Scientific Data[12]). Unfortunately, most ISA-Tab

[9] EMBL-EBI; http://precedings.nature.com/documents/3225/version/1/files/npre20093225-1.pdf.
[10] https://www.elixir-europe.org
[11] DOI: 10.1093/bioinformatics/btq415.
[12] http://www.nature.com/sdata/.

compliant tools still require the manual entry of relevant metadata and the manual linking of data sets, as well as a deep knowledge of data structures and ontologies. Thus, to date they have been rarely used by biological scientists. Although there are e orts to build more automated data capture and curation tools for specific scientific programs (e.g., FAIRDOM[13]), this area remains one of the key research topics in the field of data curation.

Challenges

Capturing sufficient curation information

Data curation advice and reference models provide guidance on the type of information that needs to be present when data is submitted for curation. However, there are no specific metadata and provenance models that implement these guidelines or link into the metadata and provenance capture process available (see §9). Furthermore, the guidelines available are focused on the long term description of the data and do not take into account the type of information required to support data reuse both in follow-on manual and automated science processes.

Selective data curation

Data curation requires significant resources, given the strongly increasing data volumes in experimental and observational facilities it is not sustainable to curate all the data ever created. Today's approaches to data selection for curation are predominantly not only manual, but also very labor intensive—capturing and assessing a range of di erent information about the data itself, its impact and its standing in the context of the complete existing collection. In large-scale data environments this approach is not sustainable. Furthermore the methods available today are focused on complete data sets and have not been applied on a much more fine-grained levels. In discussions at the workshop, domain scientists therefore identified a need for methods that would enable the automated, but selective curation of data.

Long-term data preservation

In the first instance, any data curation solution needs to ensure the integrity of the data it is managing over long periods of time. The integrity can be at risk from software errors, reprocessing errors, malicious or unintentional damage and media faults. While it is possible to combat some of these risks through data duplication, regular checks and media migration, these become cost prohibitive in extreme-scale collections as we see them at modern experimental and observational research facilities. Furthermore many of the crucial decisions along the way (i.e., when to migrate) are left to human judgment, rather than standard rules, automated checks and indicators for required actions.

Maintaining meaning over time. Experimental and observational data is only useful if it is accompanied by relevant and sufficient metadata and provenance information. However formats and accompanying tools change over time, what was once a mainstream standard can become obsolete over time, or evolve significantly due to new insights gained. Data in old formats, described by old standards becomes less useful to its designated user communities. In other cases new user communities are identified for a specific type of data, however the community is unfamiliar with the terms used to describe the data and thus cannot make adequate use of it. Today these changes are identified by humans, including the point when actions need to be taken to expand or translate specific formats. What is needed are cost-e ective means to carry out these tasks and automate their implementation.

Data curation against a background of a highly complex network of interlinked metadata standards

A subset of the challenges described in the previous paragraph are the maintenance and migration of data, metadata and provenance formats against a background of a diverse set of user groups, with their associated data lifecycles that show many links and interdependencies.

Sustainability in data curation

Today the majority of the costs involved in data curation lie in its many manual processes. Moving forward, it is expected that the increasing amount of costs will be in its storage and maintenance. New cost models are needed where approaches can be weighed and questions such as: where will long-term curated,

[13] http://fair-dom.org

searchable, accessible data archives best be hosted? Can existing and future ASCR computing and network facilities be leveraged to become part of the solution to this challenge, e.g. for a set time period? In addition one needs to consider approaches that successfully marry data integrity protection and domain specific maintenance of data, metadata, provenance, representation and reuse information.

Research focus areas

- Research into metadata and provenance models that not only support the long-term curation of data and scientific processes, but enable the active reuse of the data for identified purposes such as reanalysis, reprocessing, reproducibility, validation and background knowledge for time sensitive decision making.
- Investigate new data publication approaches that are sustainable for extreme-scale data collections, including automated, selective data curation.
- Software R&D e orts aimed at producing a sustainable approach to curation that can be applied, with tailoring, to diverse science programs across DOE SC, and that make use of existing and future ASCR computing facilities.
- New approaches to the management of collections of data products that include raw scientific data along with associated and robust metadata and provenance (see §9).
- Innovative interfaces that can support interactive access to very large data collections, possibly numerical simulations with trillions of particles or grid cells.
- R&D e ort to create automated, sustainable approaches to define, version and maintain data products (including metadata, provenance and tools) in support of publications, that have overlapping components.
- Research aimed at developing methods that enable the long-term reproducibility of curated results in support of publication verification.

10.2 Active Support for Data Validation and Reuse

A key need identified by the experimental and observational science community is the ability to support the active reuse of the curated data as part of subsequent data analysis, validation and reproducibility e orts. The community focused hereby not so much on the available information that enables reuse, but the mechanisms that can tie this data directly into automated scientific processes, potentially in time critical situations.

Challenges

Data discovery

Part of the value of a curated data archive is having the ability to find data; it is not enough to just store it somewhere. Finding data requires the ability to perform advanced searches, which implies that searches will make heavy use of metadata and provenance, as well as advances in the lexicography of search that are useful to the science community. In other words, search engine-style, text-based searches are not sufficient. Furthermore with increasingly automated research processes it is necessary to provide these discovery and access services in machine accessible form and at speeds that correspond to the requirements of the consuming services and programs.

Data curation in the context of scientific discovery

Data curation research has often viewed the process as a stand-alone activity and few studies exist that investigate its linkages to other related research activities such as: curation in support of reproducibility, validation and reuse. Challenging is hereby not only the availability of sufficient information (see §9)

to support these tasks, but the necessary software and services to support these processes as part of the automated scientific workflows.

Curating software
Curating software as might be needed for curation in support of reproducibility, is still an open research topic and the impact that new computing environments have on the reproduced results of these tools remains unstudied.

Research needs

- Advances in the lexicography of search to support common needs of the DOE SC scientific community.
- Research aimed at developing methods that enable the long-term reproducibility of curated results in support of publication verification.
- Fast search, access and analysis methods that enable the direct integration and reuse of curated data in subsequent wide ranging re-analysis workflows or utilize the information content of the curated data as background information in *in situ*, streaming or in-transit analysis and decision making processes.
- Software curation research.

Case Studies—Science User Facility Case Studies

Case Study 11

Data management, Analysis and Dissemination at the Environmental Molecular Sciences Laboratory

H. Steven Wiley, Samuel H. Payne and Matthew E. Monroe
Environmental Molecular Sciences Laboratory

11.1 Science Use Case

11.1.1 Present or Near Term

The Environmental Molecular Sciences Laboratory (EMSL) is a national scientific user facility that is funded and sponsored by the DOE's Office of Biological and Environmental Research (BER). EMSL supports BER's mission to provide innovative solutions to the Nation's environmental and energy production challenges in areas such as atmospheric aerosols, feedstocks, global carbon cycling, biogeochemistry, subsurface science and energy materials. EMSL is unique as a user facility in that it contains dozens of di erent types of instruments, including mass spectrometers, electron and light microscopes, nuclear magnetic resonance (NMR) instruments and spectrometers as well as a center for high performance computing. EMSL specializes in multidisciplinary research in which external investigators collaborate with teams of EMSL scientists to apply multiple types of analytical approaches to solve complex problems.

> Identifying, collecting and linking metadata is one of the greatest challenges that EMSL faces in analyzing and integrating research data and making it [useful] to the broader scientific community.

Because of the diversity in problem sets, research instrumentation and analytical approaches, EMSL generates highly diverse data types. Although some of our instrumentation can generate large quantities of data (e.g., mass spectrometers), data from EMSL is characterized more by its complexity than by its volume, which thus requires particular attention to the associated metadata that describes the relationship

between di erent data sets. However, because EMSL typically plays only a part in multi-partner collaborations, it usually does not have access to all of the metadata describing the data. Thus, identifying, collecting and linking metadata is one of the greatest challenges that EMSL faces in analyzing and integrating research data and making it usefully sharable to the broader scientific community. The wide variety of di erent instruments, scientific problems and approaches supported by EMSL makes it impossible to specify a single type of workflow or use case that would adequately encompass all data generation processes. Advances in specific instrumentation will necessitate the development of unique workflow and data processing pipelines to capture and analyze the attendant data. These workflows will usually be developed by the domain specialist and will be highly dependent on the specific scientific problem being pursued. However, for the data to be discoverable and useful to the wider scientific community, adequate data and metadata standards and data management systems will be required. Thus, much of the current e ort in data capture, processing and analysis at EMSL is focused on these more general needs and requirements.

In this case study, we will focus on a "typical" process that describes several common ways in which scientists interact with EMSL sta to generate data. In particular, we focus on the generation of proteomics data, which currently constitutes one of the largest research data sets in EMSL. It is also one of the best-documented types of data and is currently managed by a mature and robust data handling system. Experience in managing proteomics data in EMSL has provided us with more than a decade of experience in practical approaches for capturing, analyzing and visualizing data to support collaborative research projects. These approaches have been successfully applied to other types of data such as RNA sequencing data. It has also shown us the gaps that must be addressed to make data usefully accessible to other investigators.

General process of data generation at EMSL

The primary focus of EMSL is to understand the processes, on the molecular scale, that gives rise to complex phenomena in the chemical, physical and biological sciences by the application of powerful analytical instrumentation. This is usually accomplished by generating specific samples through an experimental protocol and then analyzing the samples with EMSL instrumentation. Although EMSL does contain a number of experimental facilities that generate samples for analysis, this is not always the case. For example, it is very common for investigators to send protein samples to EMSL for analysis by NMR or mass spectrometry. These samples will be associated with sufficient metadata to allow analysis, but not necessarily sufficient metadata to understand the significance of the results. The data from the analysis is provided to the project team that supplied the sample, which presumably has the metadata needed to make sense of the results. In practice, the analysis of the data is usually done in collaboration with EMSL scientists, but the associated metadata is still usually distributed between the user and the EMSL facility.

As an example, assume there is a project to understand how microbes break down biomass under di erent conditions in which EMSL generates proteomics data. An outside team of users would typically grow the microbes at their home institution. The user would prepare a series of samples exposed to the di erent experimental conditions, which they would ship to EMSL, requesting a specific type of analysis. EMSL sta would then process and analyze the samples based on a minimal set of metadata (e.g., microbial species, protein concentration, etc). Primary instrument data would then be used to generate derived results, based on the user needs. For example, the primary mass spectrometry data could be used to generate a list of expressed proteins, their relative abundances, how they change, etc., depending on the needs of the user and usually in an iterative fashion in collaboration with EMSL sta . Depending on the results of this primary analysis, more data processing could be performed or samples could be reanalyzed. At the end of this iterative process, a subset of the data and associated analyses is assembled into a data "package" that is provided to the user. All of the primary data and selective analyses are then stored in the EMSL Archive. Data processing is usually performed on computer clusters. The HPC center in EMSL, in contrast, is mostly used for simulations.

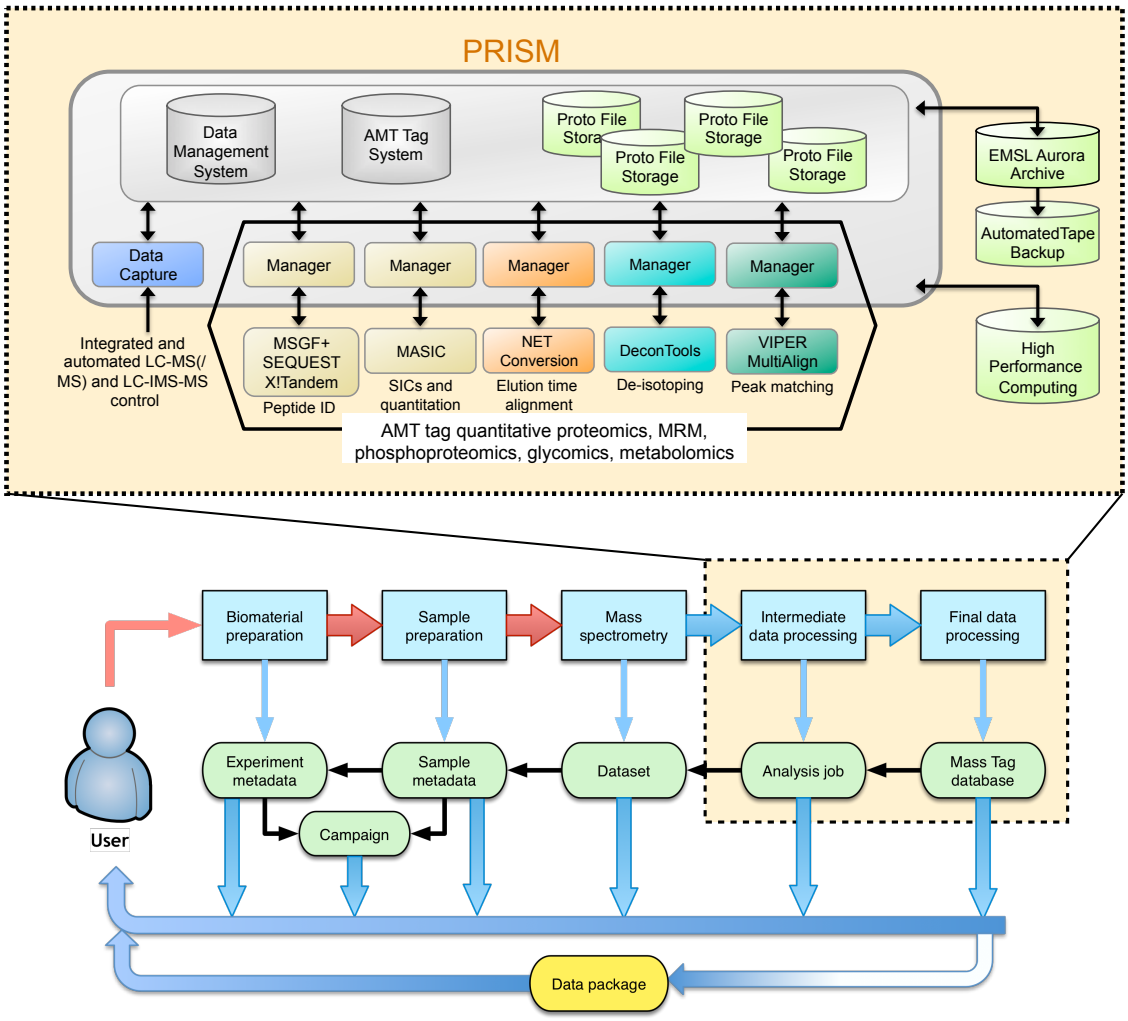


Figure 11.1: Proteomics workflow as an example user case in EMSL. A user typically prepares samples in the course of an experiment and sends the sample to EMSL. Samples are then prepared for mass spectrometry analysis by digestion and separation by high-performance liquid chromatography. The raw data files are analyzed using the PRISM data system using a series of software packages. The types of data analyses conducted depends on the individual needs of the research project. A custom set of data and metadata is then provided to the user in the form of a data package.

The above example highlights one of the more distinctive aspects of EMSL as a user facility: most EMSL users do not operate the instruments nor collect the data. Instead, EMSL sta usually perform this work. Thus, the data provided to the users constitutes a subset of all of the data collected during sample analysis and is usually highly processed. Specific capability groups within EMSL usually handle data collection, analysis and processing. The more general data needs in EMSL is understanding how to make collected data more generally useful to the research community.

Usability of collected data

In the above-described scenario, data generated by the instruments are usable, except in the rare case of equipment failure. The data might not be useful to the project generating it, but that is almost always due to a failure of experimental design on the part of the user or some unforeseen problem in sample preparation. These types of issues are outside of the ability of computational systems to o er a potential solution. The real problem, however, is that presently the data is almost never usable by anyone other than the original group that generated it. This lack of data "reusability" has many underlying causes, but this problem must be solved if making data publicly available is intended to have any useful purpose.

> The real problem . . . is that presently the data is almost never usable by anyone other than the original group that generated it . . . this problem must be solved if making data publicly available is intended to have any useful purpose. . . .

Truly reusable data requires a significant amount of associated metadata, some of which is very discipline and sample-specific. In addition, this metadata is typically distributed across multiple data storage modalities (e.g. lab notebooks, electronic spreadsheets, instrumentation software) and is generated by multiple people. Assessing and consolidating all of the relevant metadata has traditionally been extremely complex and laborious, requiring highly trained and motivated investigators. In addition, much necessary metadata is never collected because of the lack of understanding of what is required for data sharing by the primary investigator. The overall cost and complexity of metadata recording and consolidation is currently prohibitive, which is the primary reason it is rarely collected. Unfortunately, this means that the associated data cannot be easily discovered or reused.

> Truly reusable data requires a significant amount of associated metadata, some [of] which is very discipline- and sample-specific. . . . The overall cost and complexity of metadata recording and consolidation is currently prohibitive, which is the primary reason it is rarely collected. Unfortunately, this means that the associated data cannot be easily discovered or reused.

11.1.2 Future

The scope of work in EMSL is likely to continue to expand due to advances in the types and numbers of analytical technologies needed to solve DOE-relevant science questions. Together with the expansion in new data types, there will be an urgent need to improve systems to collect and manage the associated metadata, and to improve the collaborative analysis of multidimensional data sets. Both the improvement of systems and analysis will likely develop together because the needs of collaborations will require community standards and adequate metadata, but standards and metadata need to be developed around specific scientific needs. Because collaborations are usually geographically dispersed, location-independent software systems will be needed to manage group resources and analyses. At the instrumentation level workflows are likely to be similar to current ones. What will change, however, is the increased use of software frameworks to capture metadata and to support collaborative data analysis. Currently, the EMSL User Portal system maintains records on current and previous projects, including investigator information, publications and the resources allocated to each project. What it lacks, however, is specific information on the experiments and the protocols used to generate the samples analyzed within EMSL. We are currently developing a system, called MyEMSL, that will automatically collect the data and metadata needed to support advanced data analysis, visualization and sharing.

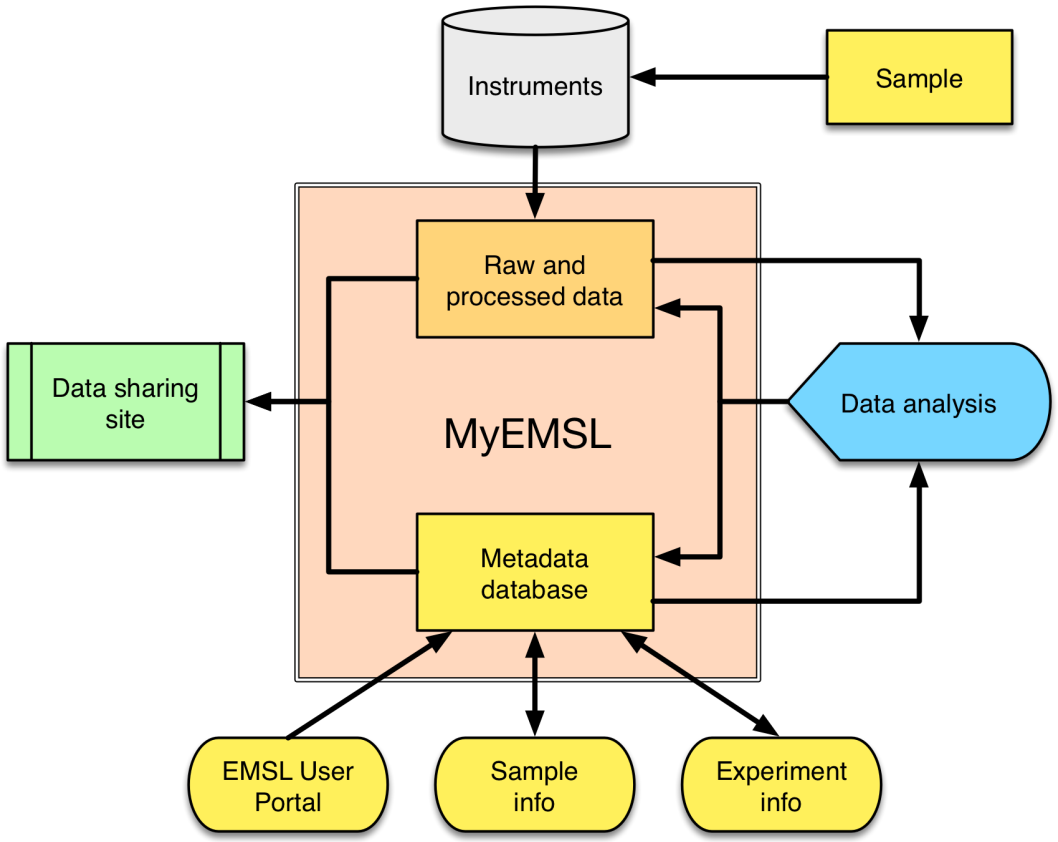


Figure 11.2: MyEMSL is a framework to consolidate data and metadata capture to facilitate data sharing and analysis. A series of software modules have been developed to automatically upload data from instruments into a central data repository. Metadata will be captured by another series of modules at both the experimental stage and the sample preparation stage and placed in a central metadata repository. The raw and processed data will be linked to the metadata through USIDs. This will allow automatic consolidation of all of the relevant data and metadata to support collaborative data analysis and sharing.

In the future, we expect to greatly increase the types of metadata captured by the MyEMSL system. We also plan to develop software modules that will capture metadata on samples sent to EMSL, or generated by EMSL sta . This data will be stored within a dedicated metadata database and linked to the project generating the samples through unique sample IDs (USID). When the samples are analyzed on EMSL instruments, the USID will be linked to the data. Secondary analysis of the data will also be linked to the USID so that all results generated by any particular sample will be clearly identified as such.

In the future, all of the data and metadata will be stored and managed using the ISA-Tab (Investigation, Study, and Assay Tabular Format) framework or a modification of that framework. Collaborative work between users and EMSL sta is expected to generate secondary analyses, models and conclusions that will also be stored back to the central data repositories. This rich set of linked information can then be pushed into data sharing sites, which will be specifically tailored for each type of data or scientific specialty (e.g., proteomics, genomics, imaging). Some of these sites currently exist as data repositories that require specific types of metadata to be associated with the primary data. For example, EMSL proteomics data is currently made publicly available through the ProteomeXchange (a.k.a. PRIDE), MassIVE, and PeptideAtlas reposi-

tories. EMSL will likely host data sharing sites for specialized data types in the future. We expect that data export to data sharing sites will be as automated as practical.

11.1.3 Data Lifecycle

Currently, the MyEMSL system captures and uploads data from selected instruments into a central data repository together with a minimal set of instrument-specific metadata. Proteomics data is managed through the DMS system, which contains all metadata as well as raw and processed data on proteomics samples. The MyEMSL system captures data from the DMS system for uploading into the central data repository. Once the primary data is analyzed, collaborators are notified and they are provided access through a data link. If modifications of standard analysis routines are requested, an appointment is usually set up with EMSL sta to discuss those modifications. This is usually done in an iterative fashion. The results of these analyses are then provided to the users, but they are not necessarily stored permanently on the DMS system.

All of the data and metadata as well as the results of analyses that are stored on the central data repository are also permanently stored in the EMSL archive and backed up by a tape system. Older data is offloaded into tape storage, which can be retrieved upon request.

11.1.4 Data-centric Requirements: Capabilities, Speeds, and Feeds

The amount of data from the proteomics facility currently represents the greatest volume generated within the EMSL facility. Although specialized equipment, such as imaging mass spectrometers can generate large bursts of data, they operate on a very intermittent basis. In addition, data capture by these instruments are handled with specialized hardware designed in concert with the instruments. In the future, the data generated by EMSL could increase by perhaps an order of magnitude to 20 TB monthly, but is unlikely to exceed this.

Processing stage	Present/Near-term	Long-term
Data acquisition rate: maximum rate(s) and monthly or annual totals	100Mbps maximum data rate; 3.6 TB monthly	1 GB/s maximum data rate; 18 TB monthly
Experiment-side processing	data reduction, preliminary analysis	data reduction, metadata collection, collaborative analysis
Real-time constraints, turnaround time from collection to result for experimental control	3 days	1 hour
Metadata/provenance capture	Metadata from instruments and automated data processing	Fully automated metadata and provenance capture together with metadata from experimental protocols and sample generation

Table 11.1: Summary of data-centric requirements for proteomics data.

11.2 Impediments, Gaps, Needs, Challenges

Our proteomics use case for EMSL can be generalized to include all environmental and biological experiments, including genomics, proteomics, transcriptomics, and/or metabolomics. For these types of data, there are a variety of impediments and challenges that hinder scientific progress, such as difficulty in interpreting complex data sets and incomplete analysis that fails to uncover the most relevant and important conclusions that could be potentially derived from a data set. Thus data is generated, but the result is sometimes not discovered. Impediments that are most detrimental are related to the issues of data sharing and collaborating in large groups, methodological transparency, and dissemination and archival capabilities.

> Impediments that are most detrimental are related to the issues of data sharing and collaborating in large groups, methodological transparency, and dissemination and archival capabilities. ... The community needs a more fluid means for sharing data and working together.

As a national laboratory, we are collaborative by design; it is the goal of our research to work with the best scientists in the Nation. However, the data volumes and complexity being generated in modern science can be intimidating to non-computational scientists. It is not our desire to make every scientist pursue a joint degree in computer science, but rather to use computational frameworks that assist them in accessing and using the data in powerful, intuitive ways.

Metadata capture and data discoverability

For data to be useful to investigators outside of the initial research group it must be discoverable and data is almost always discovered through its associated metadata. Thus, for all types of scientific data, it is necessary to first define an adequate controlled vocabulary for representing the necessary metadata and identifying where in the scientific workflow it will be captured. Unfortunately, current metadata frameworks were designed for small-scale science and are frequently inadequate for a user facility such as EMSL. For example, the ISA-Tab framework was designed as "a general-purpose framework with which to capture and communicate the complex metadata required to interpret experiments employing combinations of technologies, and the associated data files" [194].

However, just as the structure of a scientific paper (hypothesis-test-validation/refutation) was designed to communicate the results of research, rather than being a historic record of how the research was done, the ISA-Tab framework poorly corresponds to typical scientific workflows. This makes it difficult to enter data into current ISA-Tab-based software. In addition, its hierarchical, relational structure is poorly suited for use in noSQL systems.

> For data to be useful to investigators outside of the initial research group it must be discoverable and data is almost always discovered through its associated metadata.

The needs of a data sharing site are quite distinct from one designed to store or analyze data. Data sharing software must have robust features for searching for specific data types and for evaluating their relationships to people, studies, scientific fields and published results. In contrast, data storage and management software for user facilities should be optimized for the workflow used to generate and analyze data. Workflow-based frameworks are generally easier for scientists to both understand and use. Indeed most of the specialized software used by scientists is designed around their specialized workflows. Understanding

the process of how science is actually done, what information needs to be captured and where the data is generated are key issues that must be addressed to enable e ective data sharing.

> For metadata to be most useful in promoting data sharing and reuse, it must be based on standards accepted by the targeted scientific community.

For metadata to be most useful in promoting data sharing and reuse, it must be based on standards accepted by the targeted scientific community. Although these standards are available for a limited range of data types, such as genomics and proteomics data, they are missing in many cases. EMSL is currently working to generate controlled metadata vocabularies for di erent types of scientific data generated within EMSL, based on the Dublin Core specifications [195]. This e ort will be coordinated with outreach to the relevant scientific users/communities to minimize duplication of e ort and to maximize community acceptance and use.

User interface considerations

To solve the problem of metadata recording and consolidation, it is necessary to discard the idea of creating a singular software solution (e.g., electronic laboratory notebook), because no one piece of software is capable of being sufficiently flexible to easily capture all types of necessary data and metadata. Instead, EMSL is developing a framework that can support di erent software modules optimized for specific data entry tasks and automatic consolidation of this data. This way, interfaces can be designed to support the scientific workflow and the consolidation task itself is automatically accomplished by software. Currently, the MyEMSL system can upload instrument data and metadata to a central data repository or archive. Future e orts are being directed towards capturing metadata on experimental protocols and sample generation from external users and sample processing metadata from EMSL sta .

"Sample" as a key concept in a generalizable data sharing system

To support large-scale, noSQL data management systems, there must be at least one unique key-value for each data record. Our studies on the implementation of various data integration strategies have indicated that one very promising approach is to assign unique IDs to each sample that is generated by a study (i.e., experiment). All scientific studies that generate data must have samples and thus samples constitute a universal, core aspect of all studies. Scientific studies can generate multiple samples and each sample can be used in multiple analyses, but multiple studies cannot generate the same sample. It is possible for a study to not generate a sample, but to instead analyze data or samples generated by other studies. In these cases, the sample would be linked to the secondary studies through the analyses (of which there can be many per sample). In the case of a simulation or calculation, the result of the simulation (its output) would constitute the study sample.

Data sharing and analysis by large groups

Most projects that use the proteomics resources of EMSL for environmental and biological research represent a collaboration of at least 10–20 individuals. This includes a variety of domain scientists, technical sta , and students. For these collaborations, current technologies are inadequate for sharing the data between group members, especially because of their wide range of technical expertise. Many people default to email for medium size files and file transfer protocol (FTP) servers for large files on the order of gigabytes to terabytes. These file sizes and transfer mechanisms have unfortunately caused a self-selection for those who participate in data analysis. Those uncomfortable with data wrangling and statistical programming

often take a less involved role. The consequence being that the group no longer benefits from their input and insight.

> Current technologies are inadequate for sharing [...] data between group members. ...The community needs a more fluid means for sharing data and working together.

Methodological transparency

One issue with big data is that data processing can be done in a wide variety of ways. Because no two experiments are alike, there is no single standard way to process data. Furthermore, documenting methodology in a manuscript is never complete. Although some individuals have been open with their methods, such as by posting all scripts used in data analysis to a webpage, this is not required. This lack of transparency impedes scientific progress because the community fails to fully learn from each other.

Dissemination and archival capabilities

Although progress has been made in some respects to coordinate and require data publication along with results published in peer-reviewed journals, it is clear that the process of data sharing is not easy or fully accepted. One reason for this is that the mechanisms for sharing do not generally support the collaborative process that is necessary to adequately use complex data sets. For example, it is not easy or appropriate to share the chain of emails between co-PIs. Thus scientific progress is substantially challenged by the use of distinct mechanisms for private sharing and collaboration before the data is released and subsequent public data sharing after the scientific publication is released.

Solutions

The community needs a more fluid means for sharing data and working together. Fortunately, computer science software development can be used as a model for how large and diverse groups working across the globe can achieve these ambitious goals. At EMSL, infrastructure is built around the team and all work by any team member is coordinated within the infrastructure through the use of version control software. Although di erent commercial and open-source platforms have been created for this purpose (such as CVS, SVN, Git) this solution has unified software engineering for the past 30+ years. Data analysis in science is remarkably similar to software engineering in that:

> The community needs a more fluid means for sharing data and working together.

- There is a large, geographically dispersed community of individuals working together on a scope of work;
- Individuals may overlap in interest and skills, but the team generally includes many di erent types of expertise;
- The work is asynchronous; and
- Individual work must be merged into the group's work;

We therefore believe that many of the issues mentioned above as impediments, gaps, needs and challenges can be addressed by adopting/adapting the version control methodologies born from software engineering to scientific data analysis.

Case Study 12

Climate Simulation and Analysis

Philip J. Rasch[1]
Pacific Northwest National Laboratory

12.1 Science Use Case

This "use case" describes the production and subsequent analysis of simulations from a state-of-the-art global climate model being developed for the BER ACME[2] and HiLAT[3] projects that can be run at very high spatial resolution. Typical production simulations take place on DOE LCFs (Titan and Mira), or the NERSC facility (Edison), and development work and shorter simulations occur on smaller machines (institutional computing).

The ACME model is quite portable, and currently scales reasonably on LCFs. The model consists of "components" corresponding to di erent parts of the Earth system (atmosphere, ocean, etc). While the atmosphere component scales well to very high processor counts other components perform less well, and act to limit the performance. On Mira, our current application uses about 2000 nodes (4 MPI tasks/node, each task with 16 threads/task). On Titan, we have more flexibility and currently use 68000 Cores (8 MPI tasks/node, 2 threads/task). Only the atmospheric component of the model is currently capable of making efficient use of the GPUs. Our current configurations do not yet scale well to the very highest "capability" performance utilization on LCF machines unless bundled ensembles of simulations are run simultaneously.

Model calculations are performed with 64-bit arithmetic and simulation output (also called "data" hereafter, and typically written as 32-bit floats) is archived at varying frequencies ranging from hourly intervals (to characterize relatively high frequency features), to monthly averaged fields (capturing lower frequency features). Simulations of a few hours or days are made during model development and debugging, extending through centuries for production simulations that explore climate variability and responses to climate forcing agents (increasing CO_2 concentrations, land-use changes, etc). Even longer simulations are needed to explore climate responses to variations in planetary orbital evolution, or continental shifts, and to estimate long-term biogeochemical responses but these are currently too costly with this class of model, so lower resolution models with simpler physics are generally used for study of those climate features.

[1] The ACME model is developing rapidly and I intend this use case description to be an informal characterization of the modeling methodology—any errors in characterization are my own.

[2] http://climatemodeling.science.energy.gov/projects/accelerated-climate-modeling-energy.

[3] http://hilat.org.

Uncertainty quantification (UQ) plays a role in our research by exposing simulation sensitivity to uncertain parameters in process representation (e.g., clouds) in model components, and investigating non-linear feedbacks within the model.

12.1.1 Present or Near Term

Our current science problems target simulation spatial resolutions for atmospheric grid cells of an approximately 25km horizontal resolution, with vertical resolutions consisting of 20m layers near the surface, stretching to approximately 3000m near the model top at 60km, where the atmospheric density is very low and the science can tolerate lower vertical resolution. The ocean component of the model uses a somewhat higher horizontal and vertical resolution (approximately 11km near the equator to 3km in polar regions) over a somewhat smaller (about two-thirds of the planet) domain. The land component is run at approximately the same horizontal resolution as the atmosphere with fewer levels with a domain that encompasses one-third of the planet. Cryosphere components require much higher spatial resolution but occupy a much smaller areal extent.

Precise estimates of the storage requirements for model output depend on model configuration details. Several hundred monthly averaged 3D fields are archived during each month of simulation in a nominal model configuration that archives only monthly mean output. Approximately 1 TB of data output are produced per model year for the configuration specified above, and we are targeting performance of approximately five simulated years per wall clock day (e.g., 5 TB/day of model output are produced) for this kind of model for next-generation optimized configurations. Higher-frequency output of the same information at 1-hour intervals would increase the output by three orders of magnitude, easily overwhelming current computer and storage capacities, necessitating alternate strategies for high frequency data archival (e.g., archive only a subset of the total model domain (for example, for particular regions, time, or levels, for specific fields).

In addition to the *in situ* time averaging calculation, our models are just beginning to utilize more complex calculations in which diagnostics are performed on-the-fly, and summaries of information are archived. Typical *in situ* calculations include:

- Satellite and aircraft simulators (which sample the model as a satellite or aircraft would see it, to mimic sampling biases produced by real measurement systems and allow more appropriate comparison to observations);
- Calculation of probability density functions (PDFs) characterizing the frequency of occurrence of some aspect of model state (e.g., precipitation intensity);
- Compositing of high-frequency information to produce longer term estimates of features that recur with approximately repeatable higher frequencies) (e.g., diurnal variations of state variables); and

Future feature tracking may include calculations for cyclones, atmospheric rivers, fronts, etc.

Model output is written to spinning disk as the simulation proceeds, in files that contain manageable chunks of information (less than or equal to 4 GB). The output is typically written using netCDF file format.[4] Typically the output is left on spinning disk until the simulation is complete so that it can be further processed and transferred to other machines more suited for data analysis and visualization.

While more sophisticated workflows are being developed within ACME, our current workflow:

- Makes use of scripts (shell, python, etc.) to check out source code, perform model configuration, build and then run the model. The scripts have a secondary function of documenting the simulation provenance (e.g., what is the source, what platform the simulation is was performed on, how the

[4] A common well regarded file format for geophysical data that includes metadata describing the data, and provenance, `http://www.unidata.ucar.edu/software/netcdf/`.

model was configured, compiled, etc). We archive the scripts as to provide a precise specification of how the model was run that could be passed to someone else and they could reproduce the simulation. Metadata is embedded in the model output that connects the simulation to the platform, source code, and scripts used to produce the data.

- Model output is typically transferred to another machine where it can be analyzed, or regridded to produce alternate representations of the information with lower (temporal or spatial) resolution, or composited to produce climatologies (e.g., the long-term statistical average of a field. The model output may also be refactored to allow more efficient sampling for another purpose (e.g., by extracting a subset of the field). I will call this processing *stage 1 processing* hereafter.
  - While it would be desirable to perform stage 1 processing on the platform where the data is produced (or on a local machine dedicated to visualization and analysis sharing a common filesystem, like Rhea in CADES at OLCF), this seldom occurs because the LCFs are optimized for capability computing, and local machines have not yet matured to the point that they are useful. Stage 1 processing is generally performed on capacity-level platforms or institutional computing.
  - Transfer of data to the analysis platform is performed by either 1) scp with ssh keys; 2) Globus; or 3) publication using the ESGF.[5] Automating this data transfer remains a challenge (but the situation is improving).
  - scripts (shell and python) that make use of purpose-built tools for data manipulation of netCDF files are used for the regridding, compositing, and refactoring;[6] UV-CDAT;[7] and NCL.[8] As the model output is transformed, metadata is added to the output files to record the transformations that have been applied to it, which also acts as a provenance mechanism (the transformation operations are recorded, as well as fingerprints of the tools used to produce the transformation).
- Procedures have been developed to make sure that original model output and transformed data are backed up to HPSS or stored on redundant file systems on more than one platform.
- Subsequent analysis (stage 2 processing) is also currently performed on an analysis platform that is not part of an LCF. This analysis includes:
  - The use of scripts to perform routine analysis of model output to compare to previous simulations. These analysis scripts evolve over years to produce a standard analysis procedure that is used on the vast majority of model simulations.
  - The use of interactive tools (through scripts, and interactively at the command line or using a GUI with, for example R, Python, MATLAB, UV-CDAT, ParaView, NCL, etc.) to probe data rapidly.

12.1.2 Future

While science objectives and motivations are unlikely to change, next generation models and analysis frameworks are likely to di er in the following ways:

- Models will run at higher resolutions, producing more data for a fixed simulation length. Weather prediction and some Global Cloud Resolving Models are already running at these much higher resolutions (at resolutions approximately 4 or 5 times higher resolution in each spatial dimension), making the operation count (and I/O volume without changes in strategy) increase by $O(5^4)$ 600. It is difficult

[5] http://esgf.llnl.gov/.
[6] See NCO, http://nco.sourceforge.net/.
[7] http://uvcdat.llnl.gov/.
[8] http://www.ncl.ucar.edu/.

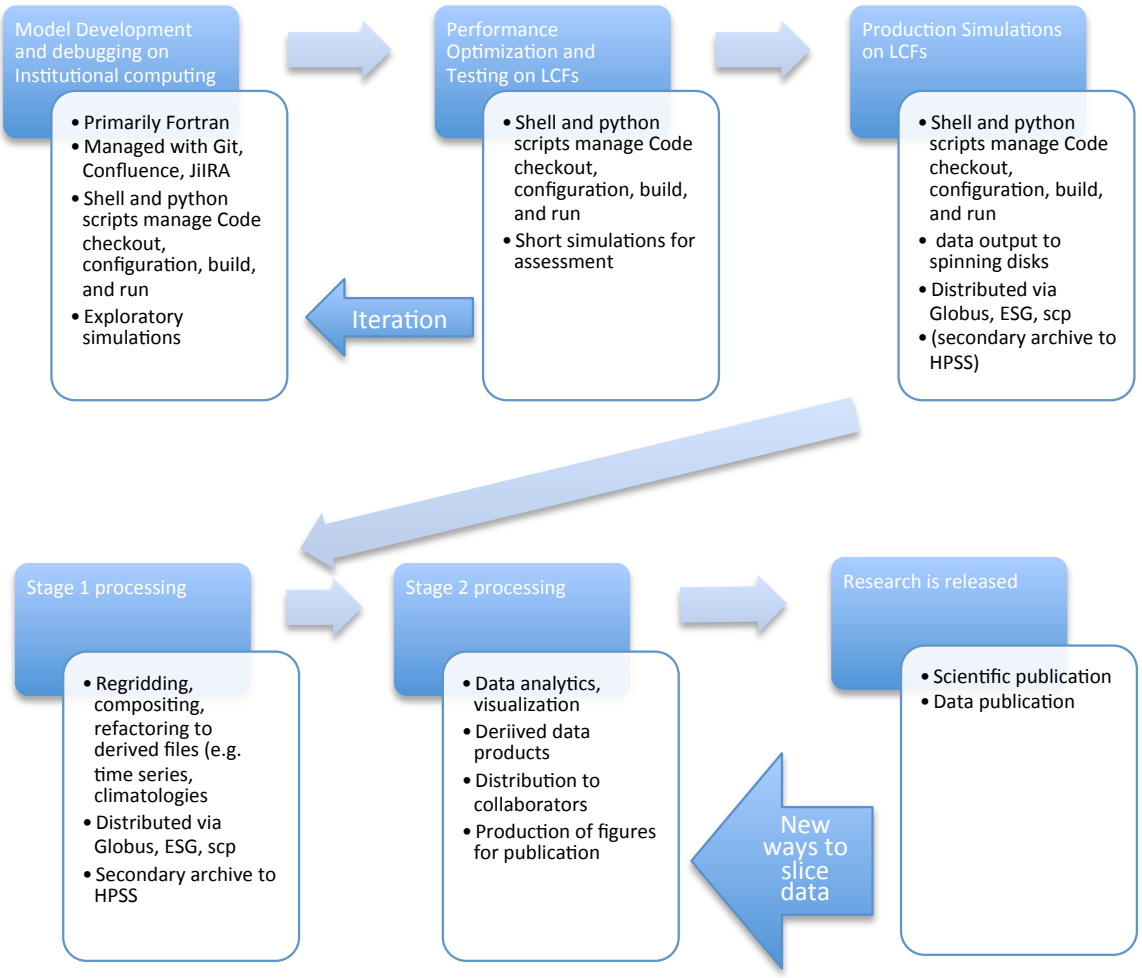


Figure 12.1: A schematic of a typical climate simulation and analysis workflow.

to estimate when future architectures/algorithms would be ready to tackle this class of simulation—it is a grand challenge, and probably unrealistic to assess for this exercise. Instead I postulate a nominal increase in resolution by a factor of two in each dimension, which would increase operation counts by a factor of sixteen and memory requirements by a factor of eight. An additional increase in complexity of representation of diabatic processes could increase costs and memory by another factor of two. So a very rough estimate in operation count and memory might be about 50 times higher than today's models.

- Next-generation models will also make better use of accelerators, and scale to much higher node, processor and thread counts, producing data at a much higher rate.[9]
- A better workflow is being developed (particularly under the ACME and SciDAC projects), which should help in automating some of the tasks that still require manual intervention (e.g., publication

[9]These issues are being addressed directly in the current ACME and SciDAC Multi-scale projects).

of data), transferring data across dedicated infrastructure for large-scale data ingress and egress.

- Conventions for model and observational metadata are evolving, improving, and becoming more standardized across modeling centers, making it possible to build tools that exploit metadata that is embedded in model output more easily.[10]
- With the increasing use of ensembles of simulations used to understand the sensitivity of the Earth system and models to characterize processes and parameters, I suspect we will begin to use tools for probing features of this multi-dimensional space more frequently, and comfortably.[11]

12.1.3 Data Lifecycle

Figure 12.1 shows a skeleton of the way climate models are developed and used. Discussion proceeds from top left to bottom right:

- In the *model development phase*, code is written, and model simulations are performed at a low resolution for short periods of time on local-, institutional- or intermediate-scale machines to provide small data sets that domain scientists, computer scientists and software engineers can use. The purpose is the development of code, debugging, development of standards for data output, and analysis tools.
- When preliminary tests look promising, the codes are moved to LCFs to undergo *performance optimization* with shorter simulations. Bottlenecks are identified and returned to the model development stage when necessary (see *iteration* arrow on Figure 12.1. Codes are then revised to improve performance. Tests with very short (1–30 day) simulations are made to confirm that solutions are insensitive to performance revisions.
- In the *production simulations on LCFs* stage, intermediate and very large calculations are performed. Intermediate-level calculations are run at either full resolution (25 km resolution, 60–70 layers) for short periods of time (1–5 year simulations), or at a lower resolution (100km, 30–60 layers) for longer time periods (decades to centuries). Ensembles are frequently run varying initial conditions, or internal model parameters to explore model internal variability, sensitivity to initial conditions, and sensitivity of model response to process or parameter variations (these are a class of *uncertainty quantification* for simulations of days to decades). Very large calculations consist of simulations for decades to centuries are occasionally run (typically perhaps a dozen per year, but for some very ambitious calculations). Data are typically retained on spinning disk to allow rapid stage 1 processing but if that is not possible, data is stored on HPSS for later processing.
- During *stage 1 processing* the data undergoes significant reduction, typically by producing climatology files that summarize some of the statistical behavior of the fields. For example, a January climatology file could be produced by compositing all of the Januarys of a century-long simulation to produce a time average value, with estimates of the time, mean and standard deviation. The result is a reduction by a factor of fifty in size of that field. Similarly, ensemble members can be averaged to characterize the ensemble mean and spread. The data can be sliced and diced in a variety of ways to allow condensed information to be retrieved much more rapidly in stage 2 processing. Sometimes the stage 1 processing occurs on machines in LCFs (such as CADES) although this has not proven particularly successful to date. Typically it must be moved, shared, and published to other locations for processing (for example, on the NERSC machines, or institutional computers).
- In *stage 2 processing*, these data sets are small enough that they can be processed on smaller machines, even as small as a desktop machine with a lot of storage (although it is easy to fill the disks). Processing is typically done with some combination of scripts (to allow easy and reliable repeatability of data manipulation) and interactive data manipulation (with a GUI, or by entering commands at

[10] See `http://cfconventions.org/`.
[11] See NDDAV: `http://www.cedmav.com/research/project/26.html`.

a command line) to probe data as understanding grows and hypotheses develop, or to produce a compelling figure to illustrate a point.

- At some point *research is released* in the form of a publication in a scientific journal, or technical note, and the data sets used to produce the publication conclusions need to be release. Alternatively data sets are often released to the community for a larger activity—for example, model intercomparison activities like CMIP5.[12]
- This cycle of research is completed by the understanding produced by the analysis leading to further scientific research and model development—then the cycle begins again.

12.1.4 Data-centric Requirements: Capabilities, Speeds, and Feeds

Processing stage	Present/Near-term	Long-term
Data rate for production simulations on LCF *standard output*: maximum rate(s) and annual total assuming we could do this every day of the year. *No attempt to optimize strategies to archive data (e.g., compression)*.	5 TB/day maximum data rate; 1.5 PB annually	Assume increase by factor of 50 over present day: 250 TB/day maximum data rate; 50 PB annually
Current strategies assume data published at the rates above will be published (shared across networks routinely).	We do not do this now, but are trying: data is currently written to HPSS.	We do not do this now, but wish we could.
Stage 1 processing: data reduction of LCF output, preliminary analysis.	Input 5 TB/day → 50 GB/day output	250 TB/day → 5 TB/day output
Stage 2 processing: routine visualization and analysis of multiple data sets output from stage 1 above.	Multiple 50 GB data sets daily.	Multiple 5 TB data sets daily.

Table 12.1: Summary of data-centric requirements.

[12] http://cmip-pcmdi.llnl.gov/.

12.2 Impediments, Gaps, Needs, Challenges

> Procedures for moving data from place to place, including tools for automating resilient workflow for orchestrating distributed data-related operations are a bottleneck (§12.2)

The previous sections have already identified many of these issues. There are a number of current bottlenecks to data analysis and visualization:

- I/O is already a signficant burden during model simulations. This increases the importance of alternate strategies for analysis, including use of *in situ* diagnostics and strategies for data compression.
- Time to transfer (identical to the sharing of) data sets across platforms is already a bottleneck, and likely to get worse as data volume increases. This is another motivation for more emphasis on strategies for data compression (both classic lossy and lossless compression techniques, and perhaps things like Principal Component Analysis or other Statistical Learning Techniques).
- Current strategies for managing (accessing, processing, keeping track of) the large number of simulations (including ensembles of simulations used in UQ) are awkward, requiring a combination of manual intervention to stratify di erent classes of simulations, and use of automated tools (to track and analyze the consequences of systematic variations in parameter settings). It would be very useful to develop procedures to systematize the characterization of simulations (perhaps developing an ontology of the kinds of model variations that we tend to work on).
- Data distribution (between machines that produce the data, and those appropriate for data analysis) is a bottleneck to climate science today that significantly impedes our scientific progress.
- There is a need today for an intermediate computing facility available to climate science that is not an LCF but is larger than the resources a single DOE lab or project is likely to be able to muster for data analysis and visualization (DAV). The DAV facility should be tied to LCFs, NERSC, and other large computing resources that produce large amounts of data through fast communication pathways to reduce the bottleneck of data sharing and publication, but focused on data analysis and visualization, archiving, and dissemination.
- procedures for some components of workflow (provenance, job submission) for producing data appear to me to be marginally adequate but could certainly be improved.
- Procedures for moving data from place to place and including tools for automating resilient workflows for orchestrating data-related actions are suboptimal and a bottleneck. They could likely be improved, and would be very nice if there were an infrastructure available that was easy to use and more powerful than our current strategies and resources.

Case Study 13

Atmospheric Radiation Measurement Climate Research Facility

Laura D. Riihimaki and Chitra Sivaraman
Pacific Northwest National Laboratory

13.1 Next-Generation Atmospheric Radiation Measurement User Facility Vision

13.1.1 Present or Near Term

The ARM Climate Research Facility, a DOE scientific user facility, provides the climate research community with strategically located *in situ* and remote sensing observatories designed to improve the understanding and representation, in climate and earth system models, of clouds and aerosols as well as their interactions and coupling with the Earth's surface. ARM operates a network of surface stations including fixed sites with long-term measurements, mobile facilities deployed for several months or years (see Figure 13.1), and an Aerial Facility with a G-1 Aircraft that makes atmospheric *in situ* measurements. Mobile and Aerial Facility campaigns are chosen through a competitive proposal process where proposed campaigns are selected based on scientific priority and practical feasibility. Ground site data comes from over 350 instruments, with another 25–30 instruments used on board the aircraft during field campaigns. All measurements and derived Value Added Products (VAPs) are stored in the ARM Archive and are available for download from ARM's webpage.[1] ARM also interacts closely with the Atmospheric Science Research (ASR) program and other science users to identify facility priorities like new instrumentation and VAP development.

> Many ... instruments ... produce very large and complex data streams ... [that] remain unmined ... and are quickly becoming the largest fraction of our data by volume ... because they must be manually interpreted by experts for data quality and meaning.

Figure 13.2 shows the movement of data from initial measurements to the ARM data archive where it can be downloaded by the user community. Currently about 18 TB of data are archived each month from the

[1] www.arm.gov.

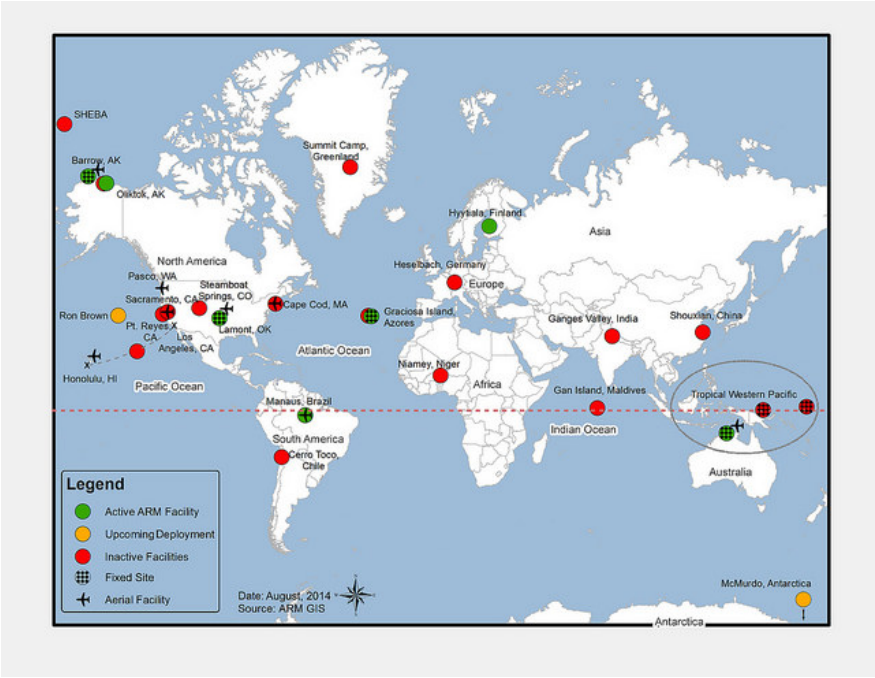


Figure 13.1: Map of fixed ARM sites, mobile facility deployments, and aerial facility campaigns as of August 2014.

over 350 instruments. Currently there are about 12 million files from 1500 di erent data streams in the ARM archive. A number of new instruments or upgraded instruments were added to the facility during the Recovery Act which promise to provide new information on some key gaps in our measurements of cloud microphysical processes, aerosol composition, and the like. However, many of these instruments (scanning radars, vertically pointing radar Doppler spectra, high spectral resolution radiometers, and Aerosol Observing System measurements) are challenging to operate and produce very large and complex data streams. Thus many of these data streams remain unmined resources, and are quickly becoming the largest fraction of our data by volume. Many of these data streams are now being used only for case studies for a few days per year because they must be manually interpreted by experts for data quality and meaning. When all of the new instrumentation is operating, it is anticipated that the annual observational data rate will be 5 PB, 10 times what is currently being archived annually. So by this estimate, and without further data product development, perhaps about 90% of data will soon fall into the category of only being accessible for small case studies by experts. Additionally, even some well-defined algorithms to retrieve useful information from these data streams are testing the limits of our current method of processing data. For example, optimal estimation retrieval methods to retrieve atmospheric humidity and cloud liquid water paths from infrared and microwave radiances call radiative transfer codes iteratively. To handle the processing, we often subsample and only retrieve 1-10% of the time steps, but if we parallelize this code and run it in an HPC environment we could run all time steps which will give better cloud statistics.

13.1.2 Future

Observations can only get us so far in fully understanding and quantifying atmospheric processes because some key quantities can not be measured, or can not be measured at an optimal spatial scale. ARM is now beginning a new approach where high resolution Large Eddy Scale (LES) models forced by observational data will be run routinely in order to fill in some of those gaps. In order to accomplish this, additional in-

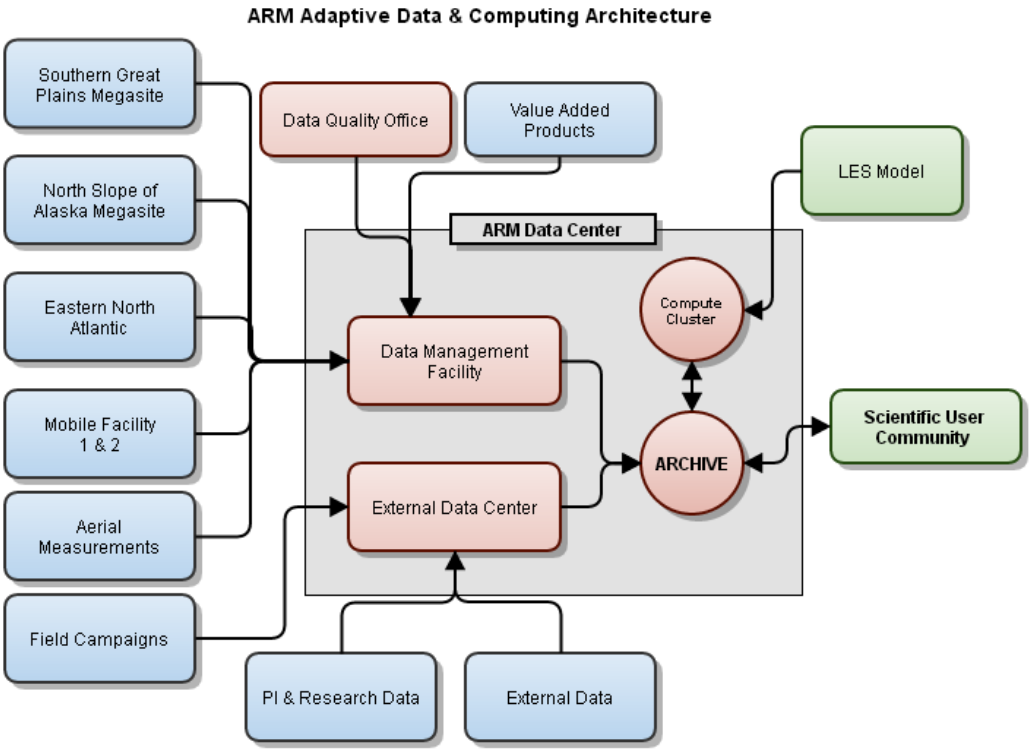


Figure 13.2: Diagram showing sources and stages of data processing from measurement to user download. Diagram courtesy of Jimmy Voyles.

strumentation has been added to the Southern Great Plains site in Oklahoma, and a team has been selected to do a two-year pilot project to develop the infrastructure needed for routine modeling. The pilot project will focus on modeling cases of shallow cumulus clouds. Several scientific motivations for this focus are the impact of shallow cumulus on global climate model temperature bias in that region, the importance of land surface heterogeneity on cloud properties, and the ability to develop statistics for parameterizations that incorporate higher order terms of the covariance between multiple parameters.

This next-generation paradigm for ARM that more closely links observations and modeling is illustrated in Figure 13.3. Box 1 shows the higher density of observational measurements at the site, along with the new transformations needed to put the data into a format for forcing the model (boxes 2–3). Modeling will be needed for data assimilation (box 5) to provide forcing for the higher resolution runs (box 4). The LES output will then be further compared to observations in an interative fashion (boxes 6–8), to refine the model (box 9) and the measurement strategy (box 10). This integrated approach will be used to build a 4D data cube that can give a more complete picture of the atmospheric state over time.

Incorporating atmospheric modeling into ARM's observational strategy will increase the computing needs. The anticipated data rate of the LES models is over 1 PB per year. LES modeling will also significantly increase the computational processing requirements for the program. One of the goals of the pilot project is to examine the costs and benefits of how much model output to store and what model settings (e.g., resolution) to use to run the model. One of the challenges, however, is to accommodate the scientific needs of a variety of users in these decisions. Initially, beta users will be identified that fit into ARM and other

BER program's scientific priorities for the LES model output. But in order to support a wider array of users, creative solutions will need to be found. For example, we will need to store the required forcing data sets and model settings in a way that users can rerun the model for the specific output they need. In addition, we will need model metrics, statistical summaries, and indices that allow users to find the subset of data of interest to them. This either requires predefining data so that a smaller subset of data can give needed information, or alternative ways to query large data sets without downloading it. A further challenge is developing the needed tools to integrate observational and model data through improved retrievals and instrument simulators. This will likely initially only be done for the subset of instruments that are expected to have the largest impact on our scientific question.

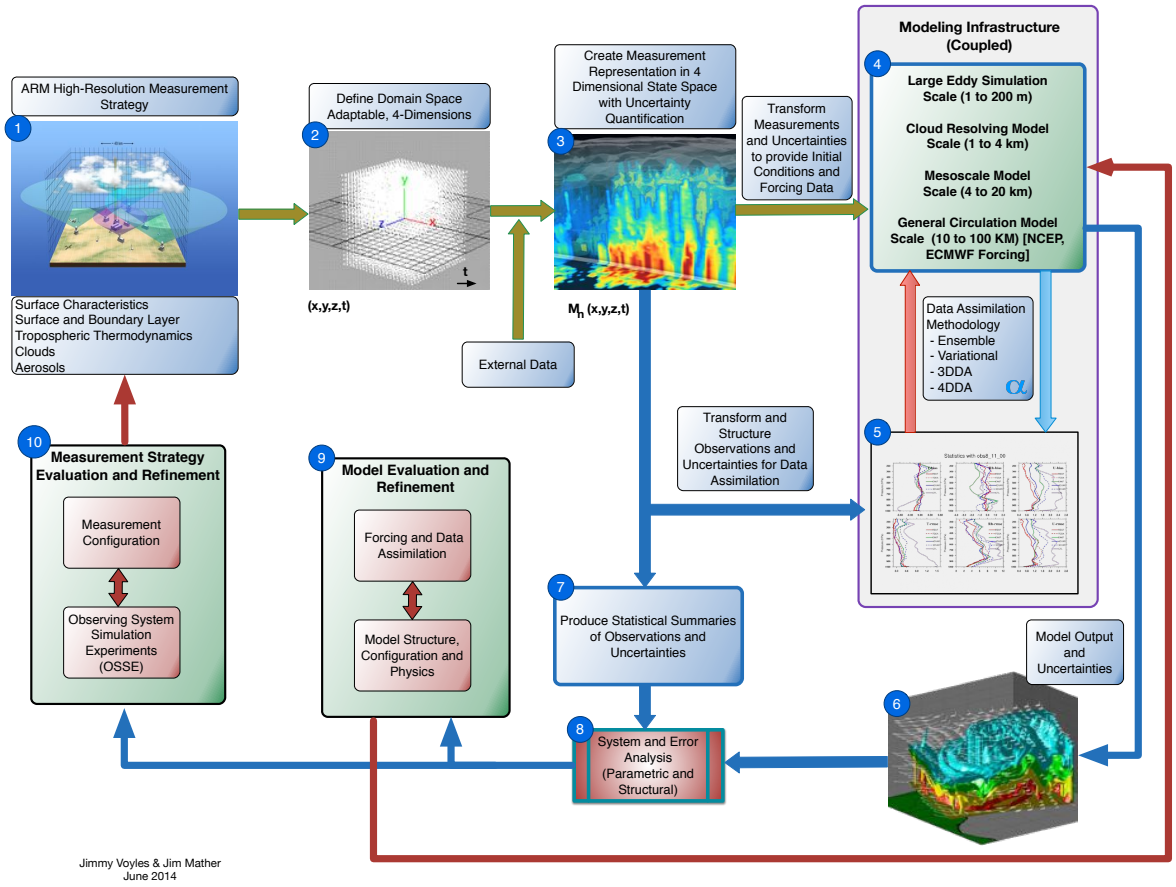


Figure 13.3: Diagram showing steps in new vision for more integration between measurement and model data.

13.1.3 Data Lifecycle

Measurements from each instrument at the fixed and mobile sites displayed in Figure 13.2 are stored on individual data loggers or computers in formats that are native to that system. That data is then transferred to the Data Management Facility (DMF) via the network for most data streams or physically on hard drives for large data streams or those at remote sites. At the DMF, the data is ingested into netCDF files following ARM standard formats.[2] Daily data is examined by students at the Data Quality Office along with a

[2] http://www.arm.gov/publications/programdocs/doe-sc-arm-15-004.pdf.

few automated quality control checks (minimum, maximum, spike). Each instrument has a mentor who is responsible for guiding the operation and interpreting the quality of the instrument's measurements. Depending on the instrument, this is a more or less accomplishable task. Some instruments are custom built by ARM or undergo significant modifications to work for the ARM purposes and take significant e ort to produce calibrated, consistent measurements.

Higher level Value Added Products (VAPs) that retrieve atmospheric parameters of interest from measurement data streams are also created autonomously at the DMF. Much of the research for developing retrieval algorithms is done by the scientific community outside of ARM. When an algorithm is deemed to be suitable for automation, and of sufficient interest to a broad community, members of the ARM infrastructure will implement those algorithms for automated processing. Both ingests and VAPs are implemented by developers, and most use the ARM data integrator (ADI) software and libraries that helps put data into a consistent format following ARM standards, transforms data into needed units and resolutions, populates databases that document dependencies, metrics, and operational status and logs, and captures provenance.

This data is then stored in the ARM archive in daily netCDF files where users can search and download data streams or individual variables from data streams from the data discovery interface (see Figure 13.4). For typical requests, the archive retrieval process extracts the user requested data from the archive, packages the data, and places the results on an ARM FTP server. A dedicated ARM 10Gbps network at ORNL processes these archive requests. Most ARM archive data is available on spinning disk, but large data streams (like radar and modeling data), and older data are stored on HPSS.

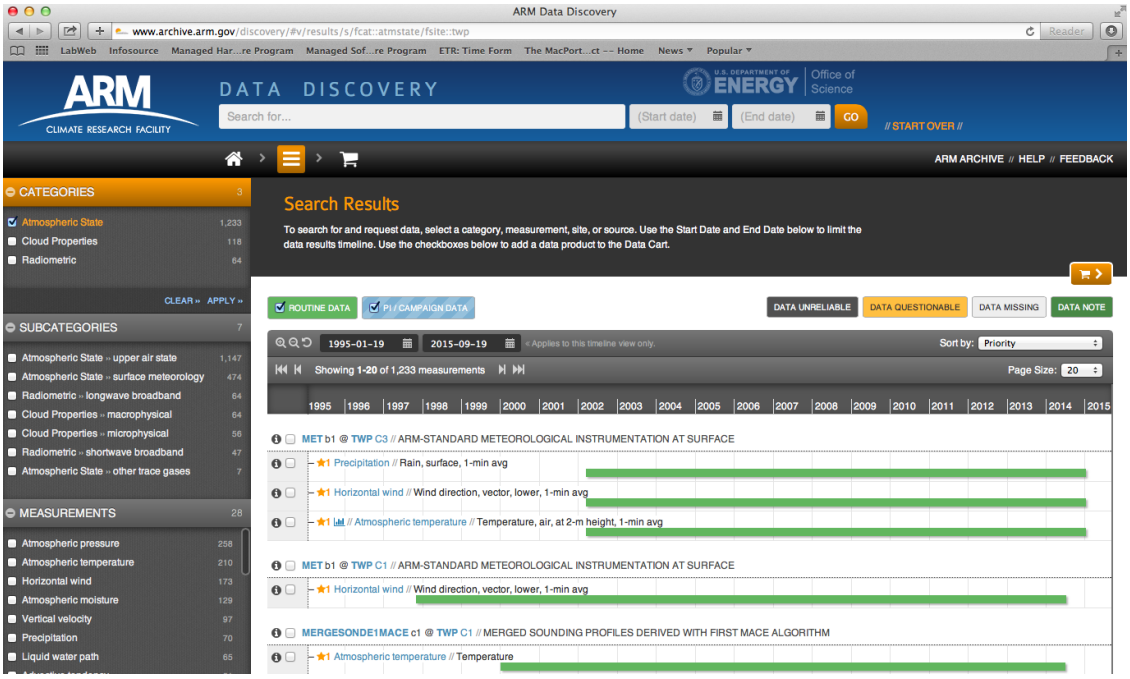


Figure 13.4: Screen shot of the data discovery web interface. Search parameters are shown in left bar.

Some large data streams, like radar Doppler spectra or scanning radars are processed on an ARM computing cluster. Additionally, discussions are underway for what resources and processes would be needed to allow scientific users to request that large data sets be moved to a computing cluster for analysis. Globus or GridFTP is used for some of the transfers between the ARM cluster and archive, but not currently between the ARM archive and other computing clusters.

The LES model output will take data forcing data sets created from ARM and external data streams using

techniques like data assimilation to provide the initial conditions for model runs. Runs will be done in an HPC environment and then some predetermined output will be stored at the archive. It will likely require a new web interface for searching data metrics and diagnostics, or updates to the data discovery interface that will allow users to subset by events of interest (e.g., convective boundary layers, shallow cumulus with or without overlying cirrus, etc).

13.1.4 Data-centric Requirements: Capabilities, Speeds, and Feeds

Processing stage	Present/Near-term	Long-term
Data acquisition rate: monthly or annual totals	18 TB/mo	5 PB observational data annually, 1 PB model data annually
Network data transfer rates by site	SGP 100 Mbps, anticipated increase to 1 Gbps in FY16; most mobile facility sites around 1–2 Mbps satellite link; Antarctica bandwidth limited to 512 kbps	May remain the same
Archive download rate	Currently about 10–15 TB per month, most individual data orders less than 100 GB	Expected to be much higher; individual LES model download rates may be of a few terabytes
Example new large data stream: radar Doppler spectra at Azores site	Five months of spectra data had volumes: 6.9 TB, 7.3 TB, 737 GB, 4.7 TB, 11 TB	This data stream will soon be collected regularly at all 5–6 sites
Searching, merging, and subsetting data	Data discovery tool can search and subset by variable, site, and measurement and will soon be able to merge some data streams into a common time using ADI	Searching and subsetting by atmospheric state, cloud type, etc.

Table 13.1: Summary of data-centric requirements.

13.2 Impediments, Gaps, Needs, Challenges

- Processing and managing large data streams:
  - We need a reliable way to compress large data streams for storage until we have algorithms that can reduce their size and give the needed information. As the science of interpreting the data progresses, we may have a need to process or reprocess large data streams for new content, but the challenge of storing this data over time is significant.
  - For reading and writing large data volumes when we process large data, like KAZR spectra, the I/O requirements are significant and sometimes prohibitive.
  - We need better parallelization of data processing when sequential data is needed. Many algorithms use data from times before or after a measurement to give context to a measurement. This makes it more difficult to process algorithms in parallel.
- Monitoring and improving data quality:
  - Ways to visualize data remotely in real time to monitor the health of an instrument is particularly a challenge at remote deployments with low bandwidth (1.5 Mbps at Oliktok and Brazil, 512 kbps at Antarctica). Part of the difficulty is a lack of resources to create visualizations of data that

will be useful for spotting problems. It is often a research e ort to understand what problems can go wrong with an instrument and how to identify them. Ways that could speed up the discovery process so that simple summary plots could be decided on more quickly would help.

  - How to achieve usable data quality (calibration, field conditions, etc.) with limited human resources to manually inspect data and develop algorithms to automate this process is another challenge. Even simple events like the recurring shading of instruments from a tree, or an instrument that is covered in direct sunlight are now often screened manually for lack of a quick system to automate these processes. There is a need for very complex automation of data quality as well. For example, it can take 20–30 hours of a trained scientist's time to get the best usable data out of three hours worth of X-band scanning radar data (about 10 GB).

- Providing ways for diverse users to access and interpret data for di erent scientific needs:
  - As data volume grows, in order for users to be able to do statistical analysis of new data streams and not be confined to a few cases, we need an interface to search and subset data without having to download it. Our plan for the short term is to create predefined, static indices like cloud classifications, LES model metrics, etc. that allow users to find cases of interest and then either download them or analyze them on an ARM computing cluster. But it would be even better to have something that would allow users to interact with the data to define their own metrics on-the-fly.
  - Increasingly, with larger radar data sets and especially new high resolution modeling output, users will need ways to transfer large amounts of data to the computing resources where they can do analysis.

Case Study 14

Advanced Light Source

Dilworth Y. Parkinson, Alexander Hexemer and Craig E. Tull
Lawrence Berkeley National Laboratory

14.1 Science Use Case

14.1.1 Present or Near Term

The Advanced Light Source (ALS), located at Lawrence Berkeley National Laboratory, is a third-generation synchrotron and national user facility that attracts scientists from around the world. The ALS has 39 beamlines as of October 2015, providing hard and soft X-rays, IR, and EUV light for imaging, scattering, and spectroscopy experiments for chemical, geological, life, material, and physical sciences.

More and more users are working on time-resolved, combinatoric, and high throughput experiments. To meet this need experimentally, synchrotrons are pushing to provide the necessary X-ray source by increasing their brightness, to build beamlines with appropriate optics and sample environments, and to work with detector developers on fast, high resolution, high efficiency detectors.

But bright sources, good optics, and fast detectors are not the only developments necessary to meet the new user needs. They must be accompanied by fast networks, high performance computers, and advanced software and algorithms. These are necessary in many cases to manage and store the large amounts of data coming at high rates, but also to reduce, process, and analyze the data to extract the useful information. Some of this computing must happen very quickly to provide feedback to users as they collect data. In other cases, more computationally intensive algorithms may be chosen—these may be slower, but they can give optimal results for subsequent analysis and publication.

The ALS has participated in two collaborations with ASCR scientists to attempt to meet users' computational needs: the Center for Advanced Mathematics for Energy Research Applications (CAMERA), which is an integrated cross-disciplinary center aimed at inventing, developing, and delivering the fundamental new mathematics required to capitalize on experimental investigations at scientific facilities; and SPOT Suite, a suite of tools developed jointly by the ALS, Berkeley Lab's Computational Research Division, the Energy Sciences Network and the National Energy Research Scientific Computing Center (NERSC), to provide ALS users access to best-of-breed data management, data analysis, and simulation tools. We will give an overview of four representative ALS beamlines that have been part of these initiatives.

- Imaging (Beamline 8.3.2, hard X-ray micro-Tomography). Scans at this beamline consist of tens to thousands of 2D X-ray transmission images ("radiographs") which are collected as a sample is rotated, generally through 180 degrees. Tomographic reconstruction yields a 3D volume with approximately 1 micron spatial resolution; in many cases, image volumes are collected every few seconds to minutes to measure dynamic processes. This beamline is used by earth scientists to study e.g. flow through porous media, by materials scientists to study e.g. material failure under strain, and by biologists to study e.g. plant and insect anatomy.
- Scattering (Beamline 7.3.3, small- and wide- angle X-ray Scattering). Small- and wide-angle X-ray scattering (SAXS/WAXS), as well as grazing incidence X-ray scattering, are techniques where the scattering of X-rays by a sample is recorded. The pattern of scattering yields information about characteristic distances within the sample on the nanometer scale, and about the shapes and sizes of macromolecules. One characteristic experiment is on organic photovoltaic (OPV) materials. Printing these materials with a specialized printer shows promise as a less expensive, more flexible way to fabricate solar cells to convert sunlight to electricity. The ALS is one of the only facilities that has been able to print and measure these materials simultaneously. By capturing an image of the solution every second for five minutes, scientists can watch the structures crystallize during the drying process
- Micro-di raction (Beamline 12.3.2). Laue x-ray microdi raction has been successfully used to probe the microstructure of materials at the (sub)micron scale. Quantities such as crystal orientation, strain/stress and defect density can be extracted from the analysis of a Laue pattern. One example of a recent experiment is with single crystal nickel-based super-alloys, the material of choice for making turbine blades in the aeronautical industry because of their excellent resistance to thermal creep and hot corrosion, and their strength at high temperature. However the cost of replacing the turbine blades when damaged can be prohibitive. Laser assisted 3D printing is the most promising alternative for repairing worn parts, as the single crystallinity needs to be maintained for the material to retain its mechanical properties. In this project, layers of Ni-based superalloys grown on a single crystalline substrate of the same material by laser assisted 3D printing under various conditions of laser power and speed, are investigated with Laue x-ray micro-di raction. Finding the conditions for the appearance of the deleterious stray grains and crack formations inside the layers are of particular interest for fine-tuning the technique. Modeling and simulation are used to assess the crystal nucleation and solidification process as well as strain distribution to be directed compared with the experimental data.
- Hybrid (COSMIC Beamline, ptychography). Ptychography is a coherent di ractive x-ray imaging method which enables x-ray imaging at a spatial resolution that is limited by the x-ray wavelength rather then the quality of x-ray optics. Images are reconstructed by a phase retrieval algorithm that acts on coherent di raction data and information known a priori about the imaging geometry. It is a scanning method, so the field of view, and hence the di raction data set, can be arbitrarily large. X-ray ptychography is in the early stages of development but is already having a very large impact in the study of chemistry and magnetism in nano-materials.

14.1.2 Future

Much of the computing at the ALS—especially prior to around 2013—was with local desktop-class machines, along with serial software developed for this platform. Much of this computing infrastructure did not integrate advanced computer science solutions. Currently and in the future, ALS needs in this area increase because of at least three changes: upgraded beamlines, new beamlines, and new approaches to the analysis and use of data after it is collected. These changes will lead to a need for adopting additional computing resources and parallelized software solutions.

- Data rate increases at existing beamlines will come from new and improved detectors, as well as from increases in flux and brightness due to upgrades in the storage ring, beamline optics, and end stations.

The increase in complexity of the experiments is enabled in part by the increases in speed, but also because of the constantly improving reliability and stability of the normal beamline components, which means that more risky and challenging experiments can be attempted.

- For the ALS, two prime examples of new data-intensive beamlines that will come online are the new ptycho-tomography beamlines, and new infrared tomography beamlines. In both of these cases, detectors will be used that can approach 10 Gbps data rates. And in both cases, the processed data will result in 5D data sets: 3D volumes that contain spectral information and which are collected as a function of time.
- There will be an increasing demand to combine data from multiple sources—not just from multiple beamlines, but from beamlines and other types of experiments, including neutron, electron, optical, and other experiments. To the extent that data becomes more widely shared and accessible across communities, we also see in the future an opportunity for a large new e ort in data mining to find patterns across data. Rather than relying solely on data you collect, you can combine results from your data with data collected by many other researchers. In many cases, this will mean the use of algorithms and questions that go far beyond the original questions and conclusions made by the researchers who collected the data.

14.1.3 Data lifecycle

We acknowledge that one barrier to collaboration between science domains and computer scientists is the lack of common terminology or representations for modeling and profiling the data lifecycle. On the other hand, it is a challenge for us to accurately portray in a simple way a characteristic data lifecycle at the ALS because it is highly experiment dependent–even for experiments at a given beamline, it can be highly variable. We will present the data lifecycle for one characteristic experiment from each of the beamlines discussed above.

- Imaging (Beamline 8.3.2, hard X-ray micro-Tomography).

  During a scan at Beamline 8.3.2, an acquisition computer saves images it receives from a camera to a Data Transfer Node, where SPOT Suite software packages each set of images (about 10 GB) and transfers them both to NERSC and to a local temporary storage server. At NERSC, preprocessing (normalization, phase retrieval, and other filtering) and tomographic reconstruction (fast analytic approaches based on Fourier transforms) is launched automatically for each data set, which results in reconstructed 3D image volumes. Users can also submit a limited number of data sets for tomographic reconstruction using iterative and model-based methods which give superior results but are orders of magnitude more computationally intensive. All results are presented to users through a web portal. Users then download the reconstructed image volumes (20–50 GB each) and carry on with subsequent steps. Data on NERSC is kept on disk for a period of a few days, until it is moved to tape, and staged back to disk on demand.

  The kinds of analysis performed on the reconstructed 3D volumes is extremely diverse, but it is common to filter and then segment structures of interest (define their boundaries); the ALS has collaborated with CAMERA to develop faster and more automated and robust methods for these steps. In many cases, segmentation is a precursor to measure porosity, or to generate statistics about the size, shape, and distribution of certain features within the volume. In other cases, a reconstructed (and often segmented and then meshed) volume is used as the input to a simulation such as reactive transport, which combines fluid dynamics, structural changes, and chemistry; these simulations using an initial volume as its starting point can be compared to the measured experimental sample during in situ time resolved experiments.

- Scattering (Beamline 7.3.3, small- and wide- angle X-ray Scattering).

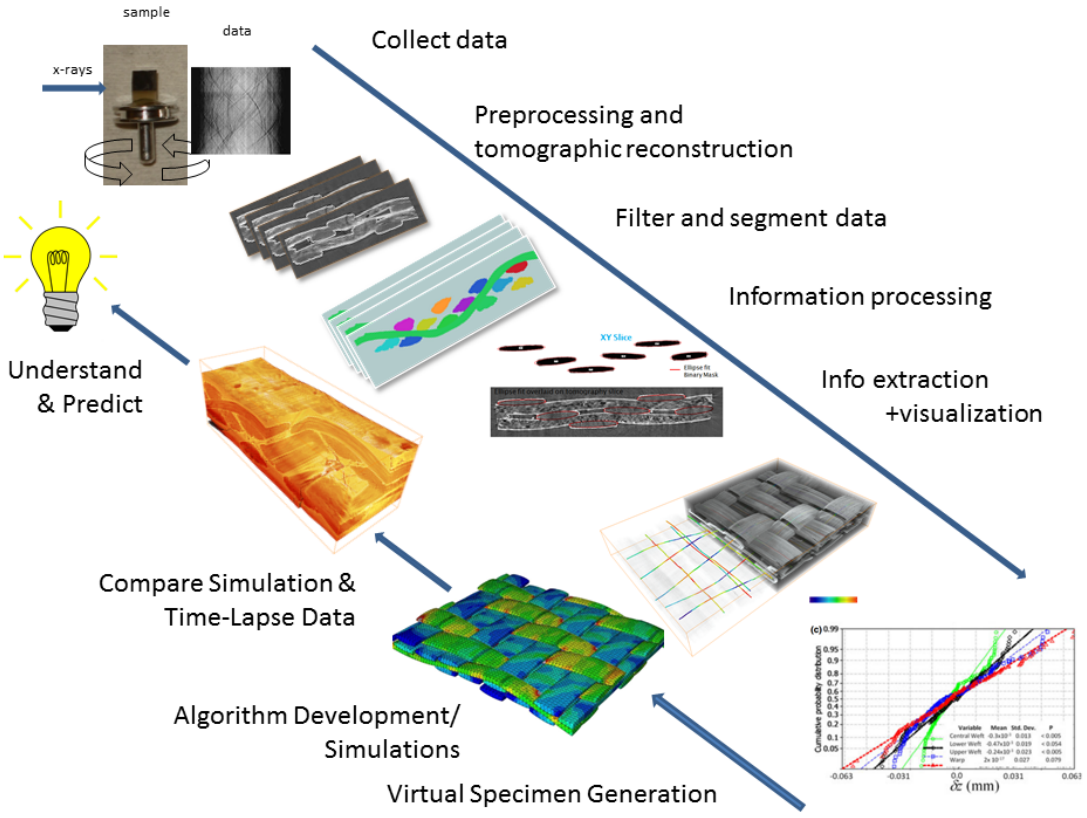


Figure 14.1: Data lifecycle for Beamline 8.3.2, from the perspective of a domain scientist (in other words, lacking details of where the computations occurred, or details about data sizes, software used, etc). Images courtesy Rob Ritchie (UC Berkeley/LBNL) and Hrishi Bale (now at Zeiss).

For one run of the organic photovoltaic (OPV) materials printing experiment described above, an attempt was made to illustrate a 'super facility' concept, with seamless integration of multiple, complementary DOE Office of Science user facilities into a virtual facility o ering fundamentally greater capability. The facilities were the ALS, NERSC, the Oak Ridge Leadership Computing Facility (OLCF) and ESnet. The SPOT Suite workflow management system running at the National Energy Research Scientific Computing Center (NERSC) was used to create a prototype data pipeline: as data was collected from an ALS GISAXS experiment, it was sent via DOE's Energy Sciences Network (ESnet) to the Titan Supercomputer at the Oak Ridge Leadership Computing Facility (OLCF) for analysis on 8000 nodes using CAMERA's HipGISAXS code, a customized high performance code that exploits advanced graphics processors and particle swarm optimization to quickly reverse engineer the sample from simulated scattering patterns based on distorted wave Born approximations. The project demonstrated the capability for researchers in organic photovoltaics to not only measure scattering patterns for their samples at the ALS and see real time feedback on all their samples through the SPOT Suite application running on NERSC, but also to see near-real time analysis of their samples running at the largest scale on the Titan supercomputer at OLCF. This allowed the researchers to understand their samples sufficiently during beamtime experiments to adjust the experiment to maximize their scientific results. Making a super facility available to users on a regular basis would have a large positive impact on the kind of work that could be done.

- Micro-di raction (Beamline 12.3.2). With the advent of fast and large-size x-ray detectors such as the

DECTRIS Pilatus hybrid pixel array detector, it has become possible to map large portion of a sample with micron step sizes within a few hours. the technique becomes particularly useful when the data generated can be analyzed in real time. We have written a Laue indexing and strain refinement code that can process multiple images in parallel. Data collected on beamline 12.3.2 of the Advanced Light Source can be transferred to NERSC automatically, through SPOT Suite. Users can then log into a web portal, where they input their desired data processing parameters, and calculations are then launched on NERSC. Results are presented within the web portal. Processing tens of thousands of Laue patterns, which previously took weeks on a desktop computer, can be done in just a few hours, so that users can get results during their beamtime and use the feedback to adjust their experiments. The use of high performance computing and fast detector technology has provided the opportunity to transition from Laue x-ray micro-di raction mapping (a few hundred data points on localized area of the sample) to a quantitative micro-structural imaging tool—1 megapixel images showing the distribution of grain orientation, phases, strain/stress and deformation inside a material.

- Hybrid (COSMIC Beamline, ptychography). Currently, di raction data is streamed from a high frame rate CCD detector, through a data transfer node and to a multi-GPU cluster during the sample scan. Data is then submitted to a preprocessing computation which removes background, filters outliers and ideally samples the di raction measurements. After preprocessing, the sample image is reconstructed by a phase retrieval algorithm that acts on the full set of di raction patterns. The parallel projection algorithm iteratively recovers reciprocal space phases and allows for a direct computation of the image via FFT. After image reconstruction, higher level analysis may proceed on a set of projections that represent a tomographic or spectroscopic image data set.

14.1.4 Data-centric Requirements: Capabilities, Speeds, and Feeds

The ALS submitted a case study to the 2014 ESnet Basic Energy Sciences Network Requirements Review.[1] This included an analysis of current and predicted data rates; those trends and predictions still hold true. We include one figure and a table that were part of that report to summarize the results of that case study.

1	Upgrade Scenario	Max Gbps	Operating Ave. Gbps	Overall Ave. Gbps
Current	Current	7.24	0.91	0.43
Current	10G LAN/WAN	7.57	0.92	0.44
Current	Detectors 5x	13.94	0.95	0.44
Current	Exposure 5x	10.37	0.95	0.44
Current	Sample 5x	7.24	1.59	0.68
Current	All 5x+10G LAN/WAN	34.56	2.98	1.31
Current+New	Current	25.84	3.22	1.81
Current+New	All 5x + 10G LAN/WAN	81.58	8.43	4.68

Figure 14.2: Based on a data rate prediction tool formulated by interviews with beamline scientists at the ALS (b1832web.1b1.gov/esnet), various scenarios can be investigated to determine expected future data rates. Predicted data rates indicate that the exponential rate of growth in network traffic seen over the last 5+ years (at least) will continue at about the same pace.

[1] *Basic Energy Sciences Network Requirements Review - Final Report 2014*: http://www.es.net/assets/Hester/RequirementsReviews/2014/BES-Net-Req-Review-2014-Final-Report.pdf.

Processing stage	Present/Near-term	Long-term
Data acquisition rate: maximum rate(s) and monthly or annual totals	10 Gbps max; 140 TB/month average (raw data)	80 Gbps max; 1.5 PB/month (raw data)
Experiment-side processing	data reduction, tomographic reconstruction, etc.	In addition, add higher-level feature extraction/identification to guide experimental system
Real-time constraints, turnaround time from collection to result for experimental control	varies among 40 beamlines from sub-second to minutes	varies by beamline, increasing numbers of beamlines will need sub-second feedback
Metadata/provenance capture	Varies by beamline	Coordinated system for capturing metadata and data from all beamlines

Table 14.1: Summary of data-centric requirements.

14.2 Impediments, Gaps, Needs, Challenges

There are a number of needs based on future plans at the ALS. Some of these have been mentioned in the previous sections. We will review them here. We note that this list has significant overlap with the report of the BES Facilities Computing Working Group from their May 2015 meeting.

- One overarching challenge is the number and diversity of light source experiments. Even for a given beamline, there are often tens of di erent types of experiments. This means that it is possible to "solve" all the problems of one user without helping the next user at all. Even when there is a potential for a given tool to benefit other users, it often takes significantly more development time to make a tool robust and easy to use to be useful to the community rather than for a single user (documentation, testing, bug tracking, message boards, outreach), and the incentive to do this extra development is often not there. On the other hand, consolidation of software would mean better software with overall less investment. One approach could be to focus on easy libraries and languages that allow relatively easy customization, rather than full environments. It will be important for these solutions to focus on parallelism in the processing and on taking advantage of emerging hardware, while facilitating running applications on multiple platforms. Another approach could be focusing on workflow tools, which could provide the vehicle that would lead to a community catalog of software libraries.
- Usability and accessibility are key concerns. It is necessary to minimize the need for facility users to have detailed knowledge of system hardware and operating systems. The ALS has been extremely successful at expanding its user base to a wide variety of science areas as well as to industry users. Many of these users will benefit from advanced computing, but they are experts in areas other than computing—writing a script is something many of them have never done. Many users also do not have easy access to computing power beyond their laptop. Even for users who do have access to more computing power, for some of the newer beamlines and for planned beamlines, it is getting to point where users cannot just download their data–their hard drive isn't big enough, and if it was they wouldn't have the computing power needed to do anything with it. In these cases, a new form of instrument combining storage, compute, data, and code is required (a "super facility" or "discovery engine").

. . . it is getting to point where users cannot just download their data—their hard drive isn't big enough, and if it was they wouldn't have the computing power needed to do anything with it.

- As data rates and experiment complexity increase, it becomes more desirable to steer the data collection, and near-real-time feedback can permit qualitatively di erent, more interactive and collaborative discovery modalities. One requirement for this will be automating and abstracting key analysis tasks, to better allocate the human in the loop—another way to think about it is the requirement to "mathematicize" more of the process (to formalize and quantify metrics that were previously qualitative). Of course, doing this would have applications beyond just real-time feedback. Another requirement to make this work will be workflows, which must help cross the boundaries of multiple data sources, computing resources, operating systems, and runtime environments to provide the necessary feedback. This is a challenge because, among other reasons, there is currently a lack of common scheduling across facilities, or a common language to define job pipeline operations across centers.
- With increasing data rates, the ALS is seeing more problems with storing data quickly enough, with how to let users access it, and with how to transfer it to users' home institutions or to their collaborators. One issue is a lack of a network infrastructure to end-users, or other issues reaching the end users—to cite one example, many users from industry have internal security policies which preclude them from using globus.org for data transfer.
- There are a number of areas in which new algorithms or analysis approaches are necessary. In many cases data is collected that is noisy or which has missing information. Some users collect large 5D data sets, and would like to visualize them with low latency or collaboratively share interactive visualizations with people at multiple locations. In other cases, users would like to combine di erent types of data from di erent instruments in a way that adds value. Many tools focus on single data sets of low dimension, so these data ensembles and high dimensional data provide a particular challenge. New visualization methods must use novel visual encoding, interactive tools for dealing with higher dimensional data, and automatic algorithms to identify salient variables across ensembles or for dimension reduction with real-time feedback.
- Ideally, any relevant data should be made available to the scientific community after some amount of time. But more than data preservation is required—proactive data curation is necessary for the data to be really useful. This will require more detailed metadata than is currently available, and it is a challenge to find ways to have automated but customizable ways to capture metadata. Data curation would mean making the data accessible in such a way that it can be searched—not just based on the existing metadata, but also based on scientifically meaningful metadata that is filled in through machine learning or other approaches. Ideally, there would be ways for data to find interested parties as it is created, rather than waiting for scientists to search for the data, which may be spread across many di erent archives. It might mean making the data accessible along with the software and computing infrastructure necessary to process the data. The benefit of curation would be to reduce duplication of e ort in data creation, but also for re-use of data for further high quality research. Another benefit would be that it could lead to more algorithms and software being made available to the community, as researchers write code that can be benchmarked and used against curated data. It is not clear who would host or pay for this data curation.

Case Study 15

Linac Coherent Light Source

Amedeo Perazzo
SLAC National Accelerator Laboratory

15.1 Science Use Case

15.1.1 Present or Near Term

The first X-ray free electron laser (FEL) to generate hard X-rays, the Linac Coherent Light Source (LCLS), began operation in 2009 and has dramatically exceeded performance expectations. This facility was created using an existing electron accelerator which limits its pulse rate to 120 Hz. It generates X-rays by amplifying spontaneous noise in the electron beam (the so-called self-amplified spontaneous emission, or SASE process), which limits the temporal coherence of the pulses.

Nevertheless, the LCLS has already had a significant impact on many areas of science, including: resolving the structures of macromolecular protein complexes that were previously inaccessible; capturing bond formation in the elusive transition-state of a chemical reaction; revealing the behavior of atoms and molecules in the presence of strong fields; observing quantum vortices in superfluid helium; and probing extreme states of matter from the structure of supercooled water to metals shock-heated to 10,000 degrees.

X-Ray range	250 to 11,300 eV
Pulse length	< 5 - 500 fs
Pulse energy	4 mJ
Repetition Rate	120 Hz

Table 15.1: Key LCLS parameters.

15.1.2 Future

The 2007 BES report *Directing Matter & Energy: Five Grand Challenges for Science & the Imagination*[1] identified fundamental open questions that underpin energy science. The report further cited the need for new

[1] The 2007 BES report can be found at: `http://science.energy.gov/~/media/bes/pdf/reports/files/gc_rpt.pdf`.

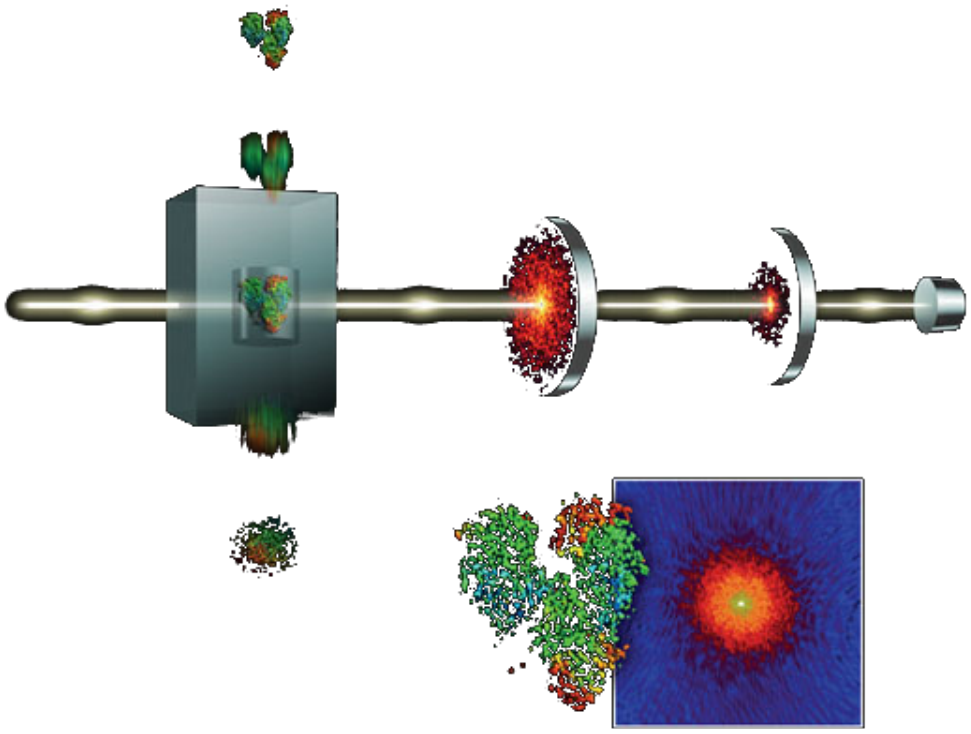


Figure 15.1: Pictorial representation of the LCLS X-ray generating di raction patterns from a sample. The ultrafast X-ray pulses are used much like flashes from a high speed strobe light, enabling scientists to take stop-motion pictures of atoms and molecules in motion, shedding light on the fundamental processes of chemistry, physics, and biology. LCLS experiments can generate up to 10GB/s sustained of science data.

observational tools and facilities to help address these grand challenges. The 2009 BES report *Next Generation Photon Sources for Grand Challenges in Science and Energy*[2] recognized specific areas of energy science where next-generation X-ray light sources would have the greatest impact. Most recently, the Report of the BES Advisory Committee's Subcommittee on Future X-ray Light Sources (2013) specifically stated: *an exciting window of opportunity exists for the U.S. to provide a revolutionary advance in X-ray science by developing and constructing an unprecedented X-ray light source. This new light source should provide high repetition rate, ultra-bright, transform limited, femtosecond X-ray pulses over a broad photon energy range with full spatial and temporal coherence.*[3]

LCLS-II represents just such an advance in X-ray laser technology and will be a transformative tool for energy science. It will qualitatively change the way in which X-ray scattering, spectroscopy and imaging will be used in the future, to observe in ways never before possible, how natural and artificial systems function, spanning multiple decades of time scales (down to the attosecond regime) and multiple spatial

[2]The 2009 BES report can be found at: `http://science.energy.gov/~/media/bes/pdf/reports/files/ngps_rpt.pdf`.

[3]The Report of the BES Advisory Committee's Subcommittee on Future X-ray Light Sources can be found at: `http://science.energy.gov/~/media/bes/besac/pdf/Reports/Future_Light_Sources_report_BESAC_approved_72513.pdf`.

scales (down to the atomic regime). LCLS-II will further enable powerful new ways to capture rare chemical events, characterize fluctuating heterogeneous complexes, and reveal underlying quantum phenomena in matter using nonlinear, multidimensional, and coherent X-ray techniques that are only possible with a true X-ray laser.

This next-generation facility will be based on advanced superconducting accelerator technology (continuous-wave radio frequencies) and tunable magnetic undulators. It will support the latest seeding technologies to provide fully coherent X-rays (at the spatial di raction limit and at the temporal transform limit) in a uniformly-spaced train of pulses with programmable repetition rates of up to 1 MHz and tunable photon energies from 0.25 to 5 keV. It will also provide coherent X-ray pulses at photon energies greatly exceeding those presently available at LCLS, up to 25 keV at 120 Hz.

15.1.3 Data Lifecycle

Data acquisition

The data acquisition system (DAQ) is the first step in the LCLS science data lifecycle. The DAQ is the set of hardware and software responsible for correctly and coherently transporting data from an experiment's cameras and detectors to offline storage. The DAQ is used to configure, calibrate, and control these devices, to read out the data, to assemble the various contributions into events tagged with the fiducial ID of the beam shot, and to write the data to disk. The DAQ also provides an online monitoring framework which allows users to analyze the quality of the data on-the-fly by snooping on the DAQ event traffic as it is sent to the data cache. Together the DAQ and the online monitoring (analysis and monitoring interface, AMI) have the ability to readout, event build, and store multi-gigabyte-per-second data streams, analyze data on-the-fly, and the flexibility to accommodate user-supplied equipment.

Each instrument—the Atomic, Molecular and Optical Science (AMO), Soft X-Ray (SXR), X-ray Pump Probe (XPP), X-ray Correlation Spectroscopy (XCS), Coherent X-Ray Imaging (CXI), and Matter Extreme Conditions (MEC)—has its own independent data acquisition system hardware and software, and all hutches may be run simultaneously.

Each experiment has a unique set of cameras, digitizers, detectors, encoders, and other devices that are required in order to accomplish the experiment's objectives. Using a set of 10–20 linux nodes per hutch connected by a dedicated 10 Gigabit network within each hutch, the data acquisition system supports the configuration, triggering, and readout of over 30 detector types that range from commercial o -the-shelf products, SLAC-built and other custom detectors, detectors along the upstream beamline, and improvised systems based on an experiment's unique requirements.

These devices can be used in any combination in each hutch; each device's readout is coordinated with the arrival of the FEL, lasers, and pulsed devices by the DAQ which distributes a trigger signal of programmable delay, width, and polarity with a known timing o set relative to the FEL beam. The DAQ system is capable of reading out 5 GB/s per instrument for all hutches except MEC which is limited to 1 GB/s, due to its lower designed data rate. CXI is capable of running two independent experiments simultaneously; consequently, its infrastructure is capable of reading out 10 GB/s.

Most devices are operated near their highest possible frame rate, and read out at the LCLS trigger rate of 120 Hz. However, the DAQ is able to acquire data from devices that read out at di erent frame rates (e.g., 120Hz, 30Hz, and 10 Hz) and associate the data with the appropriate fiducial in the event record. The data are collected by the DAQ event builder software running on a distributed system of data storage nodes and stored as complete events in the data cache. The data cache is equipped with solid-state drives (SSDs), with about 4 TB of storage per hutch and 16 TB for CXI. The data are cached in the SSDs, and the transfer to fast feedback (FFB) nodes is initiated immediately when a run is started. The fast feedback nodes can store 100-200 TB of data while awaiting transfer to permanent offline storage.

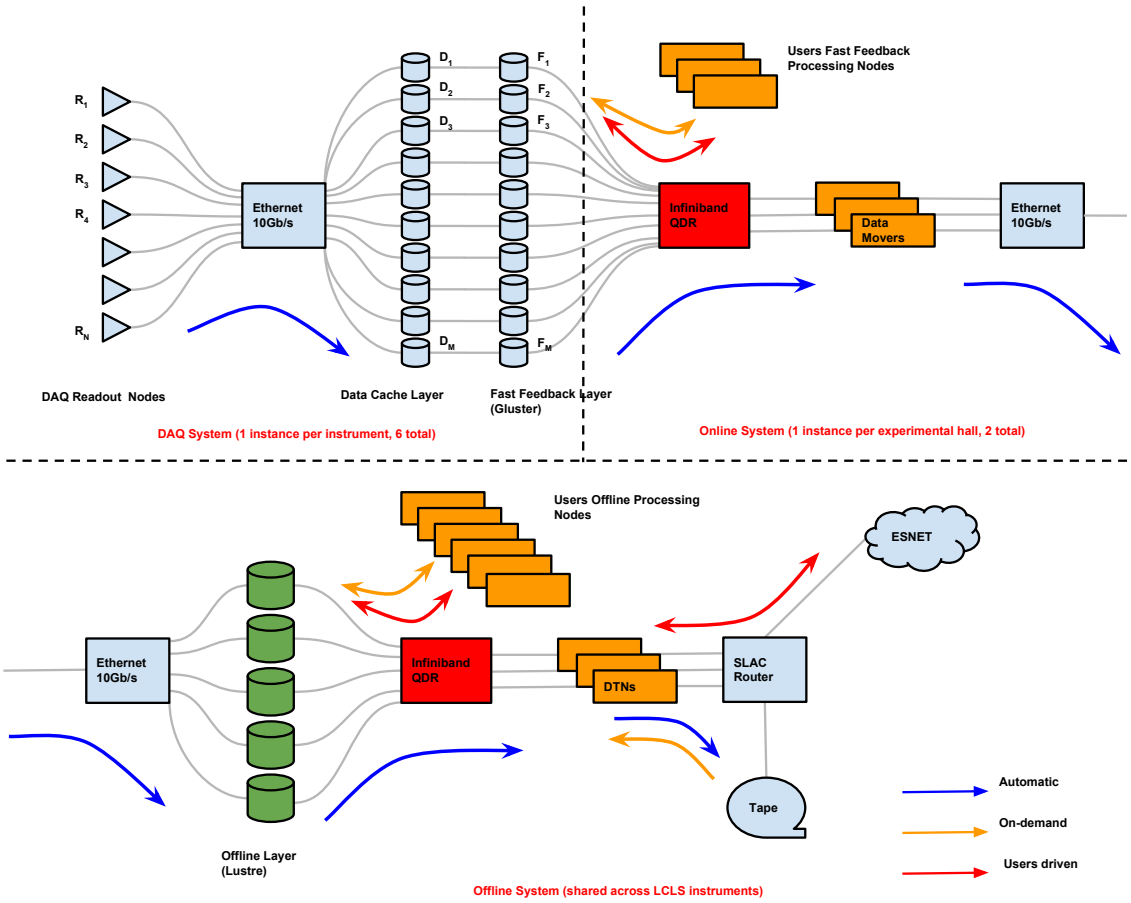


Figure 15.2: Schematic of the LCLS data flow.

The data acquisition system is controlled by the user through a GUI or through a scripted python interface that can be used to configure triggers and detectors, start/stop a run, and monitor the progress of a run. All detector devices and the set of DAQ nodes used by a specific experiment are managed through the control GUI.

Real-time monitoring

The purpose of the online AMI is to produce a user configurable, GUI-based analysis, without requiring any user coding or preparation. Monitoring and storage nodes receive each detector's data, assemble complete events, and then copy the events to shared memory where the data are promptly available for applications that allow users to perform tasks such as background subtraction, event filtering, and detector correlations. Users may also integrate their own code to perform even more sophisticated or device-specific processing. Analysis and storage farm nodes register for a multicast group to receive a fixed fraction of all events. Each hutch's processing farm typically contains over 40 CPU cores. Each node in the processing farm receives complete events containing science data from all detectors. Thus, the processing and storage load is distributed across the nodes in the farm, and each node is still capable of fully analyzing any given event. Analysis results are collected from each node for display to the operator. AMI is the default tool for online analysis and feedback.

The monitoring nodes may only monitor a fraction of the total events depending upon the number of monitoring nodes used, the data size, and the complexity of the analysis. The feedback to the user is immediate allowing analysis on-the-fly. AMI can be used on online and offline data without any coding. In the online, the processing is handled by the monitoring nodes, but may not handle every event. In the offline mode, the processing is done on the local machine and every event is analyzed. Multiple sessions of AMI may coexist so that users may monitor the data on di erent consoles and using di erent criteria. The GUI has a set of simple operations that can be cascaded to achieve a variety of monitoring measures. The monitoring automatically learns which detectors are available in the data and makes their raw data available to the user with the click of a button. All of the scalar data such as the beam energy, beamline or endstation diode values, encoder readout, Experimental Physics and Industrial Control System (EPICS) data (1 Hz sampled) associated with the event are also available. Any "posted" data, data that is the result of processing detector data for this event, is also accessible.

The data may be displayed as histograms, strip charts, distributions, and scatter plots, and it may be averaged over a configurable number of events. Scalar data can also be combined in an algebraic expression. The events included in these plots may be filtered; for example, the analysis may require that a laser shot be present in the event. The plots can be further manipulated, overlayed, displayed as a table, and saved to a text file or an image.

AMI supports single event waveform plots and image projections; these can be averaged, subtracted, and filtered. AMI has an algorithm for simple edge finding using a constant fraction discriminator. Samples can be generically manipulated by adding cursors and doing cursor math or waveform shape matching.

The online monitoring can also display data from image detectors such as commercial cameras, custom charge coupled devices (CCDs), and Pixel Array Detectors (PADs), displaying raw or custom-corrected data that includes dark subtraction, common mode noise correction, and bad pixel masking. AMI supports region-of-interest selection and masking, projections, integrals, and contrast calculations, as well as photon counting for these detectors. Other applications can run on the monitoring nodes and analyze the data received in the shared memory. Custom code can be used to plug into the monitoring framework to generate plots and contribute to the scalar data.

Data management

The main purpose of the Photon Controls and Data Systems (PCDS) data management system is to take care of the experimental data, the relevant metadata, and to make these data available for user analysis and for export to users' institutions. The system handles information produced by various sources, including:

- DAQ systems of the LCLS instruments;
- Camera images recorded by the LCLS EPICS controls system;
- Custom user data recorders run in parallel with the DAQ system in some experiments;
- Data translation services producing HDF5 files from the raw XTC files;
- LCLS users and experiment support personnel annotating data and the data taking activities during experiments and/or when analyzing data and, with some limitations, data processing and analysis activities of LCLS users.

This is a brief list of functions of the data management system:

- Maintaining a central registry of experiments;
- Implementing a safe and reliable mechanism for storing the data and metadata at various storage locations of the LCLS computing infrastructure, including disks, tape archives and databases;

- Performing data movement between original data sources, storage levels and locations within the LCLS as required by the needs of experiments and users as well as by LCLS data policies;
- Translating raw XTC files into the HDF5 format;
- Maintaining the integrity of the data throughout all stages of data movement;
- Due to the high cost of an LCLS experiment, data integrity is a paramount requirement to the system; in a scenario of limited LCLS beam time available to experiments, data losses at any level or system failures are not acceptable (experimental groups will not have "another chance" to repeat the experiments);
- Maintaining a catalog of the experimental files;
- Making the files available for user analysis and for exporting to users' institutions outside SLAC;
- Providing LCLS users with interfaces and services for managing experimental files;
- Providing Web applications for viewing and managing various metadata of the experiments;
- Providing the electronic logbook services for annotating the data taking and analysis activities;
- Implementing and enforcing LCLS data access policies;
- Enforcing LCLS data retention policies;
- Providing database and Web server services to other subsystems of LCLS and to LCLS users.

The data management system (DMS) has been designed to be an integral part of the LCLS data systems by:

- Relying upon the common computing and networking infrastructure of LCLS;
- Gathering data from LCLS DAQ and controls systems;
- Providing database and Web applications services to the LCLS DAQ systems;
- Providing data and metadata for LCLS analysis activities.
- The system also makes use of SLAC Central Computing services for archiving experimental and user data to the HPSS tape archive system.

Files are made available to users via the POSIX-compliant file systems Lustre and GlusterFS. This file storage centric approach allows users to employ their custom tools to access and to analyze the data. The primary data format for the experimental files, as they are recorded by the DAQ systems, is XTC. XTC is a homegrown binary file format allowing sequential storing and retrieval of C++ objects. XTC files can be also translated into the HDF5 format if requested by an experiment. In addition, the system:

- Has a distributed data movement architecture which supports simultaneous data migrations for all LCLS instruments;
- Is capable of handling multiple GB/s of aggregate data movements;
- Has a high level of automation using various procedures and operations;
- Uses databases and Web services as the core implementation (these services are also shared with the DAQ system);
- Has a file catalog based on iRODS;
- Enforces data integrity via checksums (presently MD5) calculated on each file at its origin (DAQ), recorded in the iRODS file catalog, and used when moving files between storage layers;

- Adopts a data security model based on UNIX groups: groups are managed by each experiment's PI or instrument support personnel.

The system includes the following databases provided via a redundant setup of MySQL servers:

- Experiment registry (experiments catalog)
- Authorization and roles (privileges)
- File catalog
- Electronic logbook
- Active experiment
- Data migration status

The Web services are built upon a redundant installation of the Apache Web servers and protected via the WebAuth/WebKDC authentication mechanism. The Web application were developed using PHP, Python and modern HTML5 technologies (JavaScript frameworks).

Data policies

LCLS doesn't impose any limits on how much data can be acquired by an experiment, or what kind of output data formats users would choose for their data processing and analysis. These choices are driven by specific needs of an experiment and what would work best in each particular case.

The data retention policies implemented at LCLS balance the storage needs of the experimental activities against storage resources available at LCLS. The core principles of the policies implemented at LCLS are:

- All data (files) and metadata (electronic logbook, etc.) of a particular experiment are by default available to the members of the experiment;
- The policies are enforced via POSIX groups. Each experiment gets its own group managed by the PI of the experiment or a person chosen by the PI;
- The raw data are kept on disk for two years with a quota imposed after 6 months. The raw data are kept on tape for 10 years.

Data analysis

The goal of the PCDS data analysis system is to provide software that allows users to:

- Quickly analyze LCLS data, both online and offline, using a common tool that is easy to use;
- Move raw or reduced data to computers of their choice in a standard format;
- Use the the same software online/offline as well as on computers not located at SLAC;
- Create central tools to perform core tasks, re-usable by many, to minimize duplication of e ort (e.g., for detector calibration);
- Create and support an additional portable data format (HDF5) so that other tools can analyze LCLS data;
- Develop and support software release tools;
- Write documentation and support users.

This work is challenging because LCLS experiments change frequently and LCLS analyses are quite diverse.

Many of the LCLS users come from synchrotron environments where data volumes are much lower, and often they are not familiar with python or C++.

The main tool supported by the offline analysis group is called psana, for Photon Systems ANAlysis. It is based on C++ and python and has been under development since 2011. The features of this software include:

- Support for both C++ and python;
- Ability to capture commonly used algorithms in reusable *modules* that can be chained together in a serial fashion;
- Support for calibrating images using standard tools;
- A new Data Description Language (DDL) that allows automatic code generation for both C++ and python data access;
- Ability to run the same software offline and online (with real-time plot display);
- Ability to analyze data parallelizing over events (up to thousands of cores).

Psana provides a framework-based analysis: this is a sequence of *modules*, written in either python or C++, that get called back sequentially for each event by the core psana code. The output object(s) of one module can be placed in an *event store* where it can be accessed by modules that follow it in the chain.

15.1.4 Data-centric Requirements: Capabilities, Speeds, and Feeds

Processing stage	Present/Near-term	Long-term
Data acquisition rate: maximum rate(s) and monthly or annual totals	1–10 GB/s maximum data rate; 1.5 PB annually	100 GB/s maximum data rate; 15 PB annually
Experiment-side processing	Detector calibration, feature extraction, histogramming, visualization	Detector calibration, feature extraction, histogramming, visualization
Real-time constraints, turnaround time from collection to result for experimental control	1–10 sec	1–10 sec

Table 15.2: Summary of data-centric requirements.

15.2 Impediments, Gaps, Needs, Challenges

The high repetition rate (1 MHz) and, above all, the potentially very high data throughput (100 GB/s) generated by LCLS-II will require a major upgrade of the data acquisition and storage system and increased data processing and data management capabilities. The main challenge will be developing high-density, high-throughput, petascale storage systems that allow concurrent access from thousands of jobs. Another critical feature is the deployment of a trigger/veto system to veto the readout process for uninteresting

events thus reducing the data throughput. Additional critical capabilities include upgrading the SLAC network connection to ESnet and expanding bandwidth and capacity of the tape archive. Specific challenges are noted here, and targeted projects to address these are outlined in the following section.

Data acquisition (DAQ): Two main modifications to the current system will be required for operating at high-repetition rates: moving the event builder from online to offline and developing the ability to aggregate contributions from multiple events in the readout nodes. These changes are required for running at 1 kHz or above, independent of throughput. The deployment of a trigger/veto system for LCLS-II will be required for large-area detectors since reading out images at full rate will not be feasible. Changes dictated by the increase in throughput are a network upgrade (from 10 Gigabit Ethernet to Infiniband or to 40 Gigabit Ethernet) and the online cache upgrade.

Real-time analysis: The LCLS experience has shown that the most e ective way to perform real-time analysis is allowing users to run their code against the data on disk (fast feedback storage layer). Fast feedback will become even more important with the deployment of a trigger/veto system for LCLS-II. The existing storage technologies are too slow for the LCLS-II fast feedback layer. Spindle-based systems will become cheaper and more dense, but not much faster or easier to manage, and they do not handle concurrency well. Solid-state-based systems will also become cheaper, but the current trend for commercial systems is to optimize IOPS (input/output operations per second) versus throughput and scalability, the key aspects for a system hosting the LCLS-II science data. In addition, current commercial systems come with a significant premium on the the cost of the flash memory, making a multi-petabyte system prohibitively expensive.

Data storage: The SLAC tape archive system is approaching limits in overall storage capacity (20+PB) and throughput. Such limits are already observed at LCLS when archiving data from on-going experiments while serving concurrent user requests to restore files from tape. Based on the current storage requirements and the estimated increase in the amount of acquired data, it is expected that LCLS-II will require between 20 and 100 PB of fast storage. Deploying and maintaining these levels of storage at SLAC would require a significant increase in the capabilities of the existing LCLS and/or SLAC IT groups. A more cost-e ective solution would be to offload part of the LCLS-II data storage to larger computing facilities like NERSC.

Data management: LCLS has developed a powerful data management system that handles both the automatic workflows of the data through the various storage layers (e.g., long-term data archival) and the users' requests through a web portal (e.g., restoring data from tape). Some aspects of the current system, such as checksum calculations, HPSS interface, and lack of prioritization, will become limitations at higher data volumes and will need to be upgraded.

Data processing: Based on the current computing requirements and the estimated increase in the amount of data to process, it is expected that LCLS-II will require between 200 TeraFLOP and 1 PetaFLOP. As with data storage, deploying and maintaining very large processing capacity at SLAC would require a significant increase in the capabilities of the existing LCLS and/or SLAC IT groups. A more e ective solution would be offloading part of the LCLS-II data processing to larger computing facilities like NERSC.

Data network: SLAC recently upgraded its connection to ESnet from 10 Gbps to 100 Gbps. The primary reason for upgrading this link is to gain the ability to offload part of the LCLS science data processing to NERSC, as current LCLS data acquisition rates are up to 5 times a single 10 Gbps link. The 100 Gbps link will not be enough for LCLS-II, and terabit capabilities will be required if LCLS relies on NERSC for processing LCLS data. In regard to the local network, Infiniband would be superior to Ethernet for building a high-throughput network for LCLS-II, especially under high congestion conditions. However, Infiniband has cost consequences in non-localized installations: it is more expensive than Ethernet to connect devices that are not within 10 meters of each other, and significantly more expensive to connect devices that are more than 300 meters apart.

Data format: The LCLS DAQ is currently writing the raw data in XTC format. Users can request that their data be translated to HDF5. The translation step will become a bottleneck in the future and LCLS-II should adopt a single data format. HDF5 is becoming the *de-facto* standard for storing science data at light source facilities, but in order to e ectively replace XTC in LCLS, a couple of critical features are required. These features, namely the ability to read while writing and the ability to consolidate multiple writers into a consistent virtual data set, are currently missing in HDF5. However, the HDF group claims these features could be added if enough resources are made available.

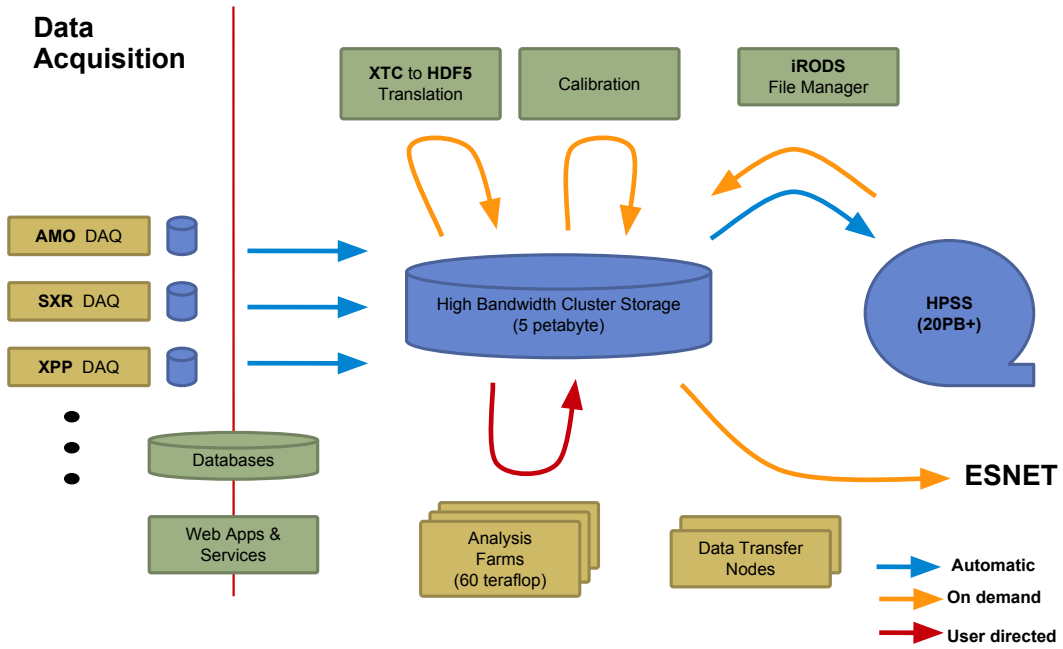


Figure 15.3: Current LCLS Data Systems Architecture.

15.2.1 Data systems projects

We currently envision an evolution of the LCLS data system where the fast feedback storage layer is built on flash memory and where the offline processing and storage capabilities can be offloaded to multiple facilities. NERSC is ideally suited to becoming one of these facilities (see Figures 15.3 and 15.4). This is a summary of the critical projects required to build a data system able to handle the LCLS-II requirements:

Event builder: Move the event builder from the online (on-the-fly data acquisition) offline (after the data are written to disk) and introduce the ability to aggregate contributions from multiple events in the readout nodes to maximize network transport efficiency. Both changes require software development only, and are incremental from what is currently in operation and required for running at 1kHz or above, independent of throughput. The online monitoring framework (AMI) and the offline framework (psana) will need to be adapted to the new paradigm.

Flash storage: Develop a custom, solid-state-based, online cache (DAQ recorders) and fast feedback storage layer (users data analysis) to solve the storage challenge. SLAC has previously worked on petascale flash-based systems and determined it is possible to build a scalable, petascale, solid-state storage by aggregating commercial o -the-shelf components. The same technology could be used to build

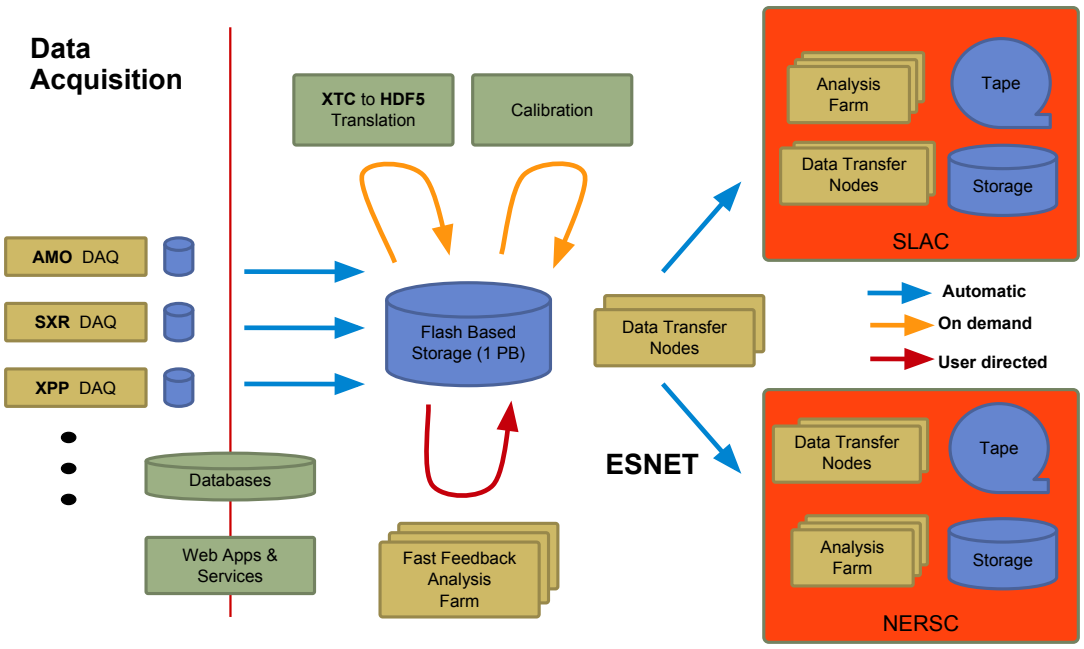


Figure 15.4: Evolution of the LCLS Data Systems Architecture.

custom recorders for both the online cache and the fast feedback system.

Upgrade local-area network: The current system uses 10 Gigabit Ethernet from the readout nodes to the online cache and to the fast feedback and Infiniband from the fast feedback to offline and within the offline nodes and the storage. SLAC will investigate introducing Infiniband or 40 Gigabit Ethernet from the readout nodes to the online cache and to the fast feedback. The final solution will be based on actual space constraints.

Add data management capabilities at high throughput: This includes upgrading the SLAC tape system, since we already see limitations when handling data from on-going experiments and concurrent users' requests to restore files from tape, and upgrading the data management framework.

Deploy new timing system: The existing event timing system is not scalable to megahertz operations, therefore a new system will be required.

Investigate a veto system: A veto signal could be delivered to the front-end electronics (EuXFEL approach), to the readout nodes, to the online cache or in the fast feedback layer. In general, a veto in the front-end electronics reduces the throughput requirements on the DAQ components, while a veto in the following layers provides cheaper/larger bu ers and more time to reach a decision.

HDF5 upgrade: Some critical features are missing from the HDF5 API. The HDF group thinks these features are useful in general, not just for the LCLS, and that they could, and should, be added to the API. The group needs additional resources to work on these features.

Increase data processing capabilities: Data centers built towards data-intensive systems could help offload the LCLS/SLAC offline computing system. General support for LCLS-II offline analysis would require more than 100 PB tape storage, a dedicated 20–100 PB of disk storage and a processing farm in 0.2–1 PetaFLOP range with an aggregate throughput to the storage above 10 GB/s per PB. These capabilities could be achieved by using dedicated resources to extend one of the large NERSC machines (Cori for LCLS-I and the next-generation supercomputer for LCLS-II). Other key requirements

are the ability for LCLS users to manage their data through the LCLS tools and workflows and, ideally, the ability to use their SLAC user account (or a federated account).

ESnet link upgrade: The ability to offload computing capabilities relies on a faster connection between SLAC and ESnet at 100 Gbps for LCLS and 1 Tbps for LCLS-II.

Case Study 16

Data for Neutron Sources at the Oak Ridge National Laboratory Neutron Sources

Garrett E. Granroth and Thomas Pro en
Oak Ridge National Laboratory

16.1 Science Use Case

16.1.1 Present or Near Term

Oak Ridge National Laboratory (ORNL) hosts two high-flux, neutron sources. One is the Spallation Neutron Source (SNS) and the other is the High Flux Isotope Reactor (HFIR). These facilities use neutrons to study structure and excitations of a wide range of materials. The primary access to these facilities is through the user program. Between internal sta and users, we service a wide range of science like: battery materials, catalysts, gas storage, strongly correlated electron systems, polymers, superconductors, biofuels, multiferroics, etc. As scattering intensities for neutrons and X-rays have very di erent scattering intensities as a function of elemental composition, these two techniques can be combined to provide optimal scientific results; thus requiring multi-modal analyses. Also in many cases, it is straight forward to compare the neutron response in a model to materials simulations. Some of these materials simulations, like Density Functional Theory (DFT) or Molecular Dynamics (MD) run on HPC platforms. Therefore there is an increasing need to simplify access to, and use of HPC resources. This includes facilitating experimental data and simulation results.

An overview of the data workflow, and the various hardware that enables it, is provided in Figure 16.1. An experiment usually begins at a planning stage which, right now, typically involves small scripts run on a user's computer or on one of the facility-provided analysis machines. Once an experiment is approved and scheduled, it is run on an instrument. Here data is collected as neutron events, meaning when a neutron is detected, its pixel ID and absolute time (to 100 ns precision) are recorded. There is roughly a week of possible local storage at each instrument for bu ering purposes. Furthermore as the events are acquired, they are streamed to the ORNL main campus where they are translated into an Event NeXus [196] file and stored on a Lustre Parallel File System (PFS). Various metadata, sample temperature, instrument parameters, etc. are also streamed down to the PFS. The PFS was chosen so that data writing activities

from the highest rate instruments do not impede the reading of other data. The streaming is handled by a publish-subscribe system known as Adara [197]. Adara was developed as a collaboration between the Neutron Sciences Directorate (NScD) and the Computing and Computational Sciences Directorate (CCSD), both located at ORNL and funded through Laboratory Director's Research and Development funds. Therefore it is an excellent example of a collaboration between Computational Sciences and Neutron Sciences to provide a robust streaming service to a BES User Facility. Besides the translation client, other clients can listen to the live stream if live access to the data is needed. One such client that has been very popular is a website [198]that allows users to track the progress of the experiments.

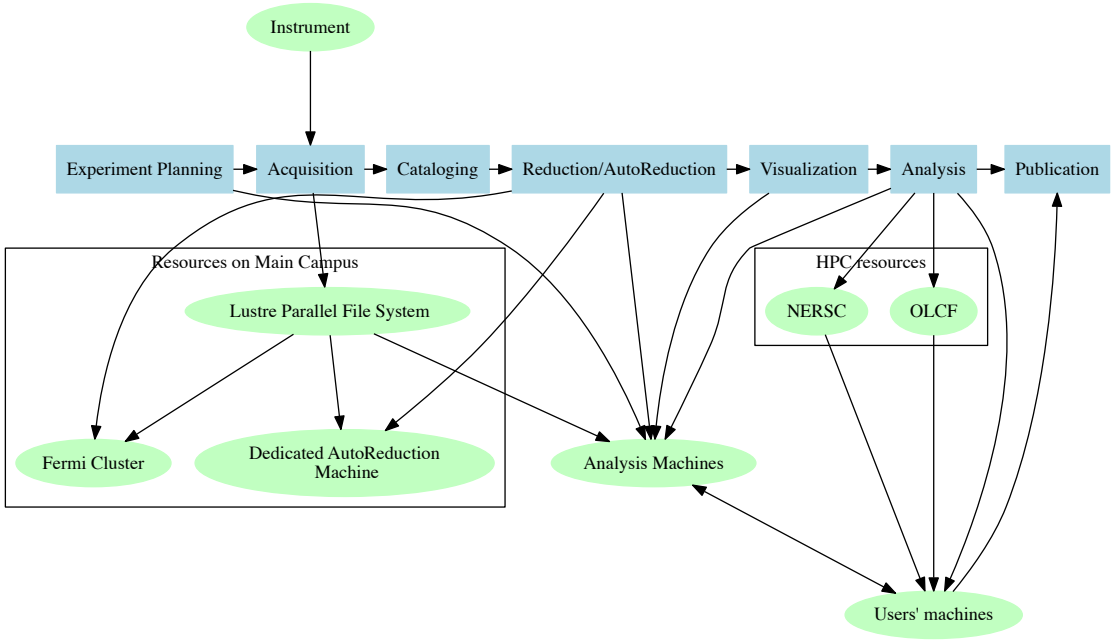


Figure 16.1: Diagram showing the current workflow for the acquisition, reduction, and analysis of neutron data at the ORNL neutron facilities. Steps in the workflow are shown in blue rectangles. The hardware, where the tasks occur, are indicated by green ellipses.

Upon closing of the NeXus file, the data is catalogued using an ICAT database [199] to allow for ease of finding data in the future. The next process in the workflow is reduction. In reduction, the data is transformed into instrument-independent units through straightforward mathematical transforms. Traditionally this process is manually controlled and launched by the user or instrument scientist. However, automated reduction is gaining popularity and most users at the SNS look at data that has been transformed in this way. By automatic we mean that the reduction process is launched as soon as the file is closed.

There are three types of hardware where reduction may occur. Manual reduction, and automated reduction that does not require a lot of resource, usually occurs on one of several analysis machines at the SNS. There are two analysis machines per beamline and another nine that are shared between all instruments. The standard configuration for these machines is 64 cores and 512 GB of RAM. Since the first neutrons at the SNS, reduction has been performed on such a cluster, although the size of these machines has grown, since the beginning of the facility. For reduction that can benefit from parallelization, a 100-core cluster called Fermi, and co-located with the PFS, can be used. Also auto-reduction for instruments that require large compute resources are performed on a dedicated auto-reduction machine. Some of these dedicated machines are located at the facility and some are on the main campus. The PFS is mounted on all of these machines to provide access to the collected data and for a place to record the reduced data. In this way data is transferred back and forth to the facility on an as needed basis. Reduction is handled by the Mantid [200]

software framework.

After the data is reduced it may be visualized and then analyzed, or just analyzed, depending on the technique. The reduced data varies greatly in size—it can be as small as a few megabytes to as large as 300 GB. We provide the analysis machines to allow the users to visualize and analyze their data. Currently most of the analysis codes were designed to run at previous generation facilities. In the cases where the data is smaller, some users do their analysis on the cluster and some simply transfer it, via ssh and Ethernet, to their personal computer for analysis. The traditional software, when run on instrument configurations at the SNS, requires as much as 300GB of RAM. Most users do not have access to such computational resources at their home institution. Furthermore the time to download such a reduced data set makes use of the remote analysis resources more appealing to most users. Although some who have access to ESnet or Internet2 at their home institutions would like the ability to use faster data transfer methods.

Visualization currently occurs most frequently with traditional software such as Horace [201] or Dave-Mslice [202] which are Matlab [203] and IDL [204] based codes, respectively. The field is in a transition to using software that leverages ParaView [205] for visualization. This is occurring to leverage parallel resources to visualize volume renderings of data sets that are hundreds of gigabytes in size.

An example of this transition is shown by a slice through a 4D data set from a neutron spectroscopy measurement on $(Tl,Rb)_2Fe_4Se_5$ shown in Figure 16.2. This slice was processed in ParaView using the jet colormap. This data was originally processed with Horace [206]. F. Samsel visualized this data using a

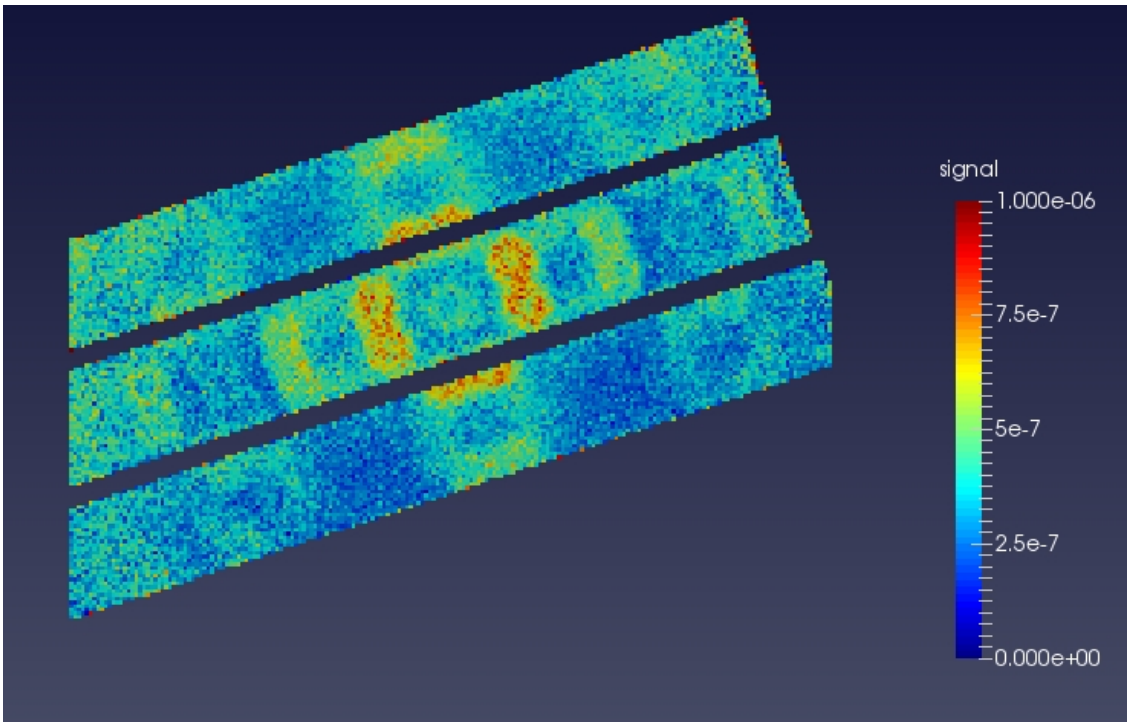


Figure 16.2: A slice of a 4-D neutron spectroscopy data set as viewed from ParaView

di erent colormap (Figure 16.3), The result shows more detail, especially near the level of the background (di erences in blue shades). Therefore gains in understanding should be possible by providing better tools for manipulating color maps and education in what makes a good colormap.

There is growing interest in using computationally intensive techniques like DFT or MD in analysis. For these cases we utilize facilities at the OLCF, NERSC, or other resources the users themselves provide. As

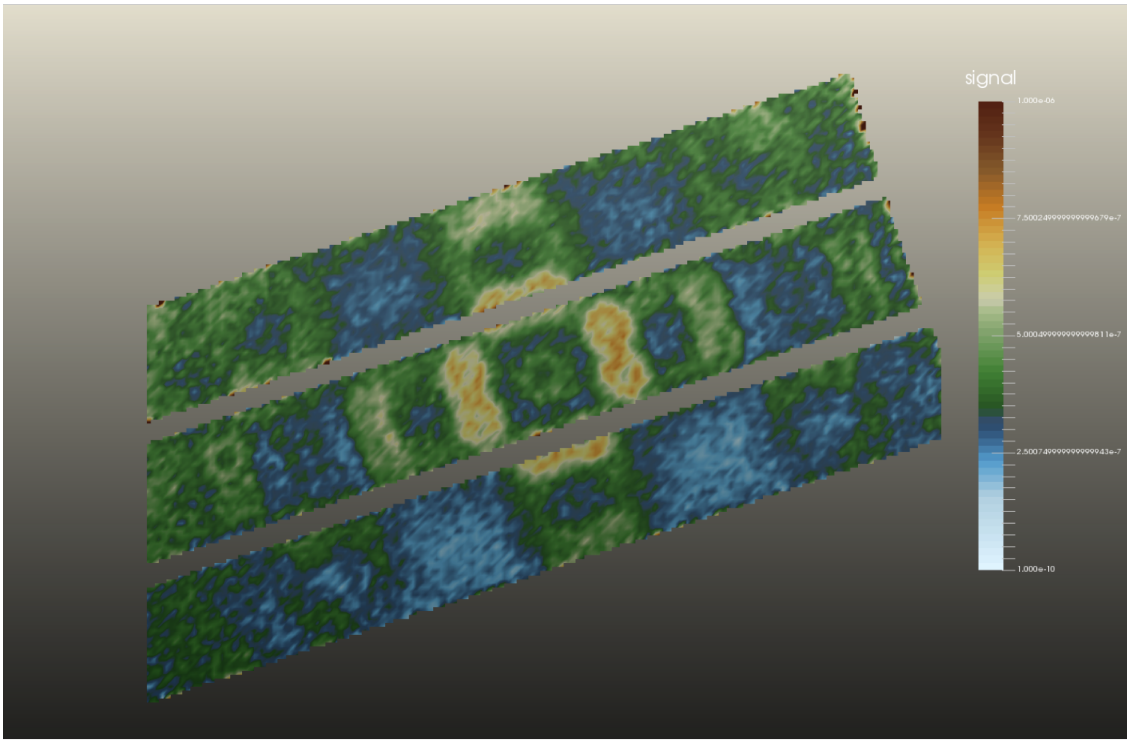


Figure 16.3: A slice of a 4D neutron spectroscopy data set as viewed from ParaView.

this grows, we expect to need more access to these resources. Right now we also leave it up to the user to get the results of the simulation o the HPC resource and to decide where to compare their data to the simulation. Right now we have no limitation on the size of the results that users can upload to the system. This works because demand is currently low and the process for uploading data requires slow links and the user's computer to be in the middle. However this is not scalable and a long-term solution for storing and moving the appropriate reduced data, and the required simulation results must be found.

> Right now we leave it up to the user to [manually perform simulation-experiment data comparison/analysis, along with associated manual data movement] . . . However this is not scalable and a long term solution for storing and moving the appropriate reduced data, and the required simulation results must be found.

Similarly many studies co-analyze X-ray data taken at one of the users facilities. However this data again must traverse the user computer for comparison. Here this works because the vast number of use cases are passing reduced data sets that are small. We can envision the need to pass around large data sets, say tomographic reconstructions of neutron and X-ray imaging data. Our current system would support only a limited amount of this comparison.

> . . . our only archival process right now is that provided by the published journal.

The output of this process is typically paper production. Notice that this task comes from the users' computers, and our only archival process right now is that provided by the published journal. The current

typical output per instrument is roughly one paper per every two experiments. The net result of this is approximately 400 papers per year for the facility.

16.1.2 Future

Figure 16.4 shows a diagram of the workflow and the associated hardware in the future. For the most part acquisition and reduction will see little change. One exception arises from a time-of-flight imaging beamline that we expect to bring online within the next 5 years. It may require significantly more computing at the beamline so that we can send computed tomographs to the PFS rather than raw data. This instrument may merit an expansion of the Adara streaming system as the data rate is higher. A synergy with the X-ray user facilities may also be found as the data is a stream of images, like the data produced by X-ray detectors, rather than a stream of events. Second we expect almost all reduction to move to automated resources so the analysis machines are used more for analysis. This is a trend we are seeing and we expect it to continue.

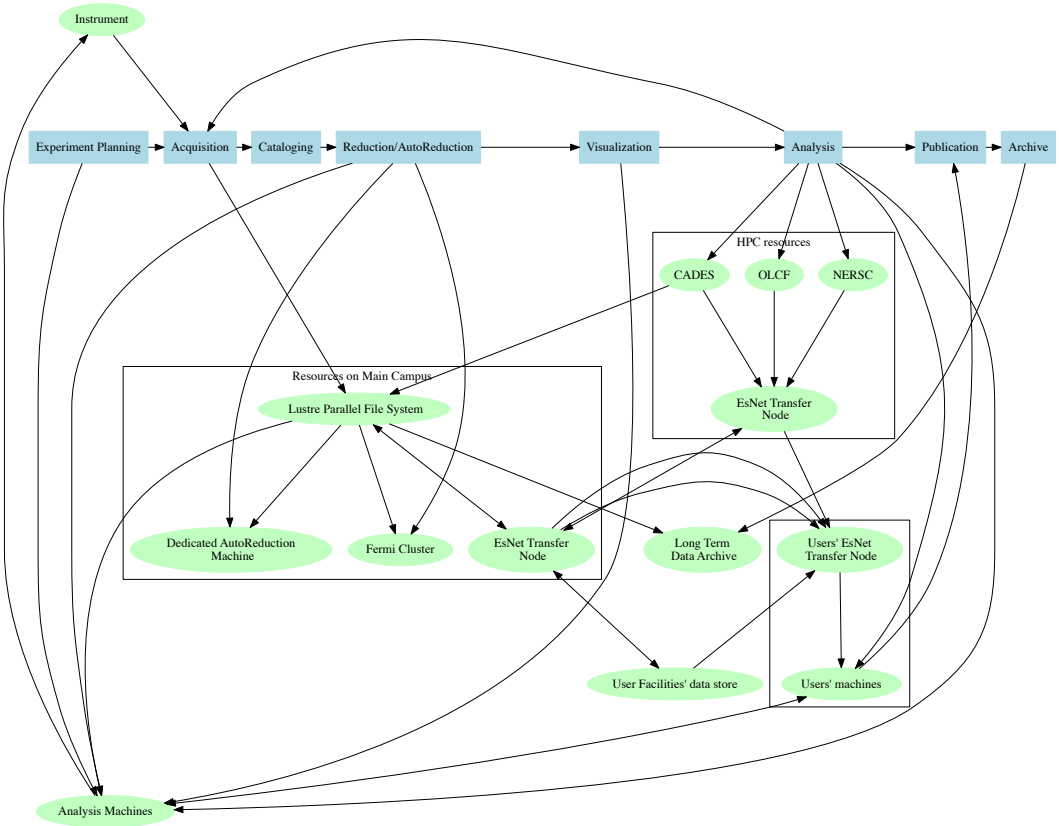


Figure 16.4: Diagram showing the expected future data workflow. Again blue rectangles indicate steps in the process and green ellipses indicate hardware used to execute the workflow. Note that feedback has been implemented between analysis and acquisition. Also more high-speed data transfer capabilities are included.

The diagram changes significantly as we get to analysis. First note that feedback has been introduced from analysis back to acquisition. This is a growing request from instrument sta . The most straightforward

idea is to check simple quantities like the statistical significance in a region of interest on the detector bank. Such processes can be handled by the analysis machines; thus the hardware connection between the analysis machines and the instrument. However one could envision simultaneously running a simulation requiring HPC and acquiring data at the same time. Acquisition and or simulation, would move to the next step based on some heuristic comparing the two data sets.

Another aspect of analysis is we expect to make more use of HPC resources. Note that the CADES resource at ORNL is introduced into the HPC box. This is a moderate level computing resource that is well matched to many of the physical problems studied by neutron scattering. It will also be tied directly to the PFS allowing for the direct comparison of data. We are testing this model with DFT calculations of molecular vibration over the next years and expect it to expand to more techniques. Other HPC resources like amazon web services [207] or Microsoft azure [208] may also be used as well.

Introduction of ESnet connections is also planned to help future analysis work. This will allow easy exchange of data between multiple facilities and HPC centers. Also if users have access, they should be able to download their data and or simulations.

Finally we have explicitly spelled out a long-term data store. With increased data rates from new instruments and additional detector banks on other instruments, we expect we will no longer be able to keep all data and all reduced data on the PFS. Some appropriate longer term resource is required. Furthermore there is interest, especially among the theoretical community, of having access to data after the current researchers have published. Such access needs to ensure that enough metadata is stored that the data can be analyzed appropriately. We have the ability to store raw data and the reduction script used to get from raw data to reduced data. However we need an electronic logbook to capture the reason why certain aspects of analysis or reduction were performed. Furthermore this notebook needs to be archived with the data so this subsequent access is useful. Providing more access to the data, in a manner that can be used by more scientists, will improve efficiency, increase the impact of the science, and result in more papers per experiment.

Providing more access to the data, in a manner that can be used by more scientists, will improve efficiency, increase the impact of the science, and result in more papers per experiment.

From a di erent perspective there are gains to be had by introducing advanced mathematical methods in several aspects. Using multi-modal data to constrain a model is one example. The current state of the art is co-refinement of a structural model using neutron and X-ray di raction data with programs like GSAS [209] and Fulprof [210]. More generally a Baysean approach of optimizing a model from whatever experimental data is available, is the future. Next we want to use the optimized algorithm on the optimal platform via an architecture that allows the user to assemble novel workflows using simple scripts. To this end we need a library of HPC-optimized routines (numerical integration, Metropolis Monte Carlo, optimizers like Dakota [211], etc.), each with a python interface, that can be assembled in this workflow. For visualization we have made a first pass at adaptive binning procedures in the Mantid Software [200]. This uses rectangular bins [212]. But as the user community becomes more familiar with this, technique we will look into tessellated bins [213].

16.1.3 Data Lifecycle

Currently raw data is generated during the acquisition and the researchers are generally done with it once the publication has been completed. Our policy is to keep raw data indefinitely, and we have no specific policy about reduced data. However, with our highest rate instruments now online, we may have to rethink this strategy within the next two years. There is also interest in rethinking our policy that the user controls

access to the data in perpetuity. Researchers, beyond the traditional experimentalists would like access to compare standard data sets to models. We are currently storing more metadata with the reduced data than ever before in the history of the community. We are also now storing the algorithms used to convert a raw data set to a reduced data set. However to allow the fullest use of data sets to the broader scientific community will also require tracking analysis procedures and to some degree tracking notes.

16.1.4 Data-centric Requirements: Capabilities, Speeds, and Feeds

Processing stage	Present/Near-term	Long-term
Data acquisition rate: maximum rate(s) and monthly or annual totals	500 MB/s maximum data rate; 0.3 PB annually	5 GB/s maximum data rate; 1 PB annually
Experiment-side processing	data reduction, traditional analysis	data reduction, traditional analysis, electronic notebook meta data
Real-time constraints, turnaround time from collection to result for experimental control	5–15 minutes	5 sec
Metadata/provenance capture	This is done for most automated experiment variables and for reduction	capture appropriate analysis metadata and notebook style information.

Table 16.1: Summary of data-centric requirements.

16.2 Impediments, Gaps, Needs, Challenges

In summary, we see the following items as challenges that will need to be overcome.

- Archiving the appropriate data and reduced data or data and procedure to reduce data;
- In the aforementioned archive, include the appropriate metadata so it can be utilized by researchers beyond the group that acquired it;
- Implementing a facility to enable data sharing and reuse of already acquired data;
- Transferring large data sets between scattering user facilities, between HPC resources and scattering facilities, and providing high speed access to it for users that can take advantage of the high speed data links.

Case Study 17

Data and Analysis Requirements in Scanning Probe and Electron Microscopies

S.V. Kalinin, A. Belianinov, A. Lupini, S. Somnath, E. Strelcov, and S. Jesse

Oak Ridge National Laboratory

17.1 Science Use Case

Scanning probe (atomic force, scanning tunneling, etc.) and scanning transmission electron microscopies now form the mainstay of nanoscience by providing capabilities for the local characterization and manipulation of matter on the nanometer and atomic scales. Until very recently, development of these techniques was based on the synergy of instrumentation platforms (stability, noise, or environment), improved probes and detectors, measurement modalities, and mathematical tools for the extraction of materials-specific parameters from imaging data. However, in almost all cases the information provided to the researcher is in the form of 2D images and (in the last decade) 3D spectroscopic imaging data sets, whereas full information flow within the microscope was unavailable for the operator and end users and the internal analytics (required for feedback systems, for example) was limited. Furthermore, the generated data sets are usually manually sub-selected for subsequent detailed analysis by the researcher, further limiting the information generation capability of these tools and obviating data re-use. This paradigm di ers significantly from that in large scattering and synchrotron facilities for example.

Here, we analyze the information aspects of probe and electron microscopy imaging as a first step for developing systematic solutions for full data utilization and reuse in imaging. Given the traditional gap between the fields, we perform the analysis separately for scanning probe microscopy (SPM) and ((Scanning) Transmission Electron Microscopy, (S)TEM, and elucidate commonalities when possible. We also note that SPM operates with scalar (single data stream) excitation and detection signals over a 2D scanning area, whereas STEM allows for much broader variability of detection schemes (0D, 1D, 2D) and scanned areas (2D or 3D). Hence we analyze SPM first and STEM second.

The SPM group at the Center for Nanophase Materials Sciences (CNMS) is actively working on the development of scanning probe microscopy techniques for probing bias-induced (ferroelectric polarization switch-

ing, electrochemical reactions) and thermal (glass transition, melting) transformations at the nanoscale. In these experiments, the SPM tip focuses an electric or thermal field in a small (5–30 nm) region of material, inducing local transformations. In parallel, measured dynamic strain, resonance frequency shift, or quality factor of the cantilever (piezoresponse force microscopy, electrochemical strain microscopy) or tip-surface current (conductive atomic force microscopy, AFM) provides information on processes in the material (polarization, domain size, ionic motion, second phase formation, melting) induced by local stimulus. In the future, the detection strategies can include microwave, Raman, focused X-ray, electron microscopy, and other high-bandwidth local (approximately on the order of 10 nm and below) structural and chemical probes.

The uniqueness of this approach is that the transformation can be probed in material volumes containing no or single individual extended defects, paving a pathway for studying **phase transformations and electrochemical reactions at the single defect level** (as opposed to volume averaging for typical materials science methods; compare to the impact of molecular unfolding spectroscopy in biomolecular chemistry), the target of crucial importance for materials science to link a defect structure to its functionality.

> These [future] studies require drastic improvement in their **capability to collect and analyze multidimensional data sets**, well beyond the state of the art (2D imaging or 3D spectroscopic imaging) in the field.

The hardware platforms for these studies can be realized on 30,000+ SPMs worldwide and necessitate a classical development path of minimizing by noise level, improving drift stability, and introducing proper chemical and thermal environments. However, these studies require drastic improvement in **capability to collect and analyze multidimensional data sets**, well beyond state of the art (2D imaging or 3D spectroscopic imaging) in the field. This can be demonstrated as follows:

- The spatial scanning necessitates data acquisition over 2D dense grid of points;
- The probing local transformation requires sweeping local stimulus (tip bias or temperature) while measuring the response;
- All first order phase transitions are hysteretic and hence are history dependent. This necessitates types of first-order reversal curve studies, e ectively increasing the dimensionality of data (e.g., probing Preisach densities);
- First-order phase transition often possess slow time dynamics, necessitating probing kinetic hysteresis (and di erentiating it from thermodynamics) by measuring a response as a function of time; and
- The detection of force-based SPMs necessitates a probing response in a frequency band around resonance (since resonant frequency can be position dependent and single-frequency methods fail to capture these changes).

These simple physical arguments illustrate that complete probing of local transformations necessitate 6D (space × frequency × (stimulus × stimulus) × time) detection scheme, as compared to the 1D molecular unfolding spectroscopy. To date, we have realized 5D and tentative 6D detection schemes (first order reversal curves, time relaxation within hysteresis loop methods). The development of these techniques is illustrated in Table 17.1. Figure 17.1 shows the evolution of information volume for selected scanning probe microscopy techniques since their invention.

(S)TEM and associated focused ion beam (FIB) microscopies spectroscopies are well established, robust imaging tools that have proved to be powerful for the visualization of structure and functionality of materials with atomic resolution [214, 215]. The ultimate goal of localized imaging and spectroscopy is to observe and quantitatively correlate structure-property relationships with functionality—by evaluating chemical, electronic, optical and phonon properties of individual atomic and nanometer-sized structural

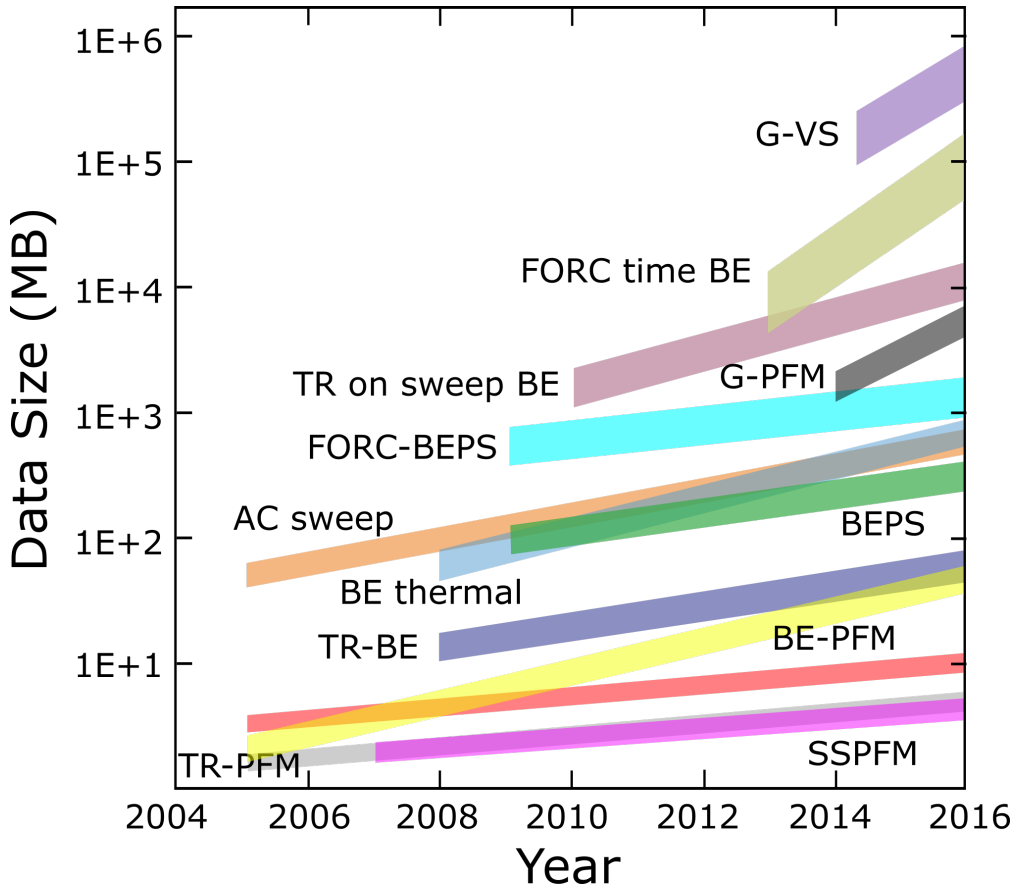


Figure 17.1: Evolution of information volume in multidimensional scanning probe microscopies.

elements [216]. Historic improvements in the underlying instrument hardware and data processing technologies has allowed for the determination of atomic positions with sub-ten-picometer precision [217, 218], which enabled the visualization of chemical and mechanical strains [219], and order parameter fields including ferroelectric polarization [220, 221, 222, 223] and octahedral tilts [224, 225, 226, 227, 228]. Ideally, complete studies have to be performed as a function of global *stimuli*, such as the temperature or uniform electric field applied to the system, as well as local *stimuli* that are induced by additional probe or ionic interactions [229, 230, 231]. Furthermore, this technical combinatorial instrumentation challenge is exacerbated by a wealth of extracted information at both global and local scales necessitating a drastic improvement in *capability to transfer, store and analyze multidimensional data sets.*

17.1.1 Present or Near Term

Traditionally, data analysis, storage, and distribution e orts in the scanning probe, electron and ion microscopy domains are the responsibility of an individual sta or user; with whatever limited data analysis knowledge and capability available to them. Only an insignificant fraction of data is analyzed (based on the initial screening during the acquisition process), and a fraction of analyzed data is published and becomes available for community-wide examination. Many analytical tools are custom designed, and are rarely traceable. The delayed use or re-use of data is common for a single PI, but is highly unlikely outside the groups by the broader community. At the same time, the value of complete utilization and re-use of data are obvious, and both domain-specific and synergistic opportunities enabled by it can be easily

envisioned.

> Traditionally, data analysis, storage, and distribution e orts in the scanning probe, electron and ion microscopy domains are the responsibility of an individual sta , or user; with whatever limited data analysis knowledge and capability available to them. Only [an] insignificant fraction of data is analyzed (based on initial screening during acquisition process), and [only a] fraction of analyzed data is published and becomes available for community-wide examination.

In the last year, the SPM group at CNMS, in association with OLCF and the Institute for Functional Imaging of Materials (IFIM) made significant strides in implementing an HPC infrastructure called BEAM (Bellerophon Environment for Analysis of Materials) [232]. BEAM enables instrument scientists to leverage the integrated computational and analytical power of ORNL's CADES platform with HPC resources at the OLCF and at the National Energy Research Scientific Computing Center (NERSC) to perform near-real-time, scalable data analysis via a web-deliverable, cross-platform Java application. At the core of this cluster-based computing system is a web and data server located in CADES that enables multiple, concurrent users to securely upload and manage data, execute materials science workflows, and interactively engage analysis artifacts. BEAM's long-term data management services utilize CADES large-scale storage system and enable users to easily manipulate remote directories and upload or download new and processed data in their private data storage area as if they were browsing on a local workstation. Additionally, this framework accepts custom data analysis algorithms (developed by mathematicians, computational scientists, and materials scientists) in order to enable user-defined workflow needs; and allows post-authentication, "push button" execution of dynamically generated workflows on multiple DOE HPC platforms and CADES compute clusters (a.k.a., the "DOE HPC Cloud").

> Many analytical tools are custom designed, and are rarely traceable. The delayed use or re-use of data is common for a single PI, but is highly unlikely outside the groups by broader community.

Currently a custom set of algorithms, that are broadly described as multivariate analysis, curve fitting and image feature recognition, are being implemented on BEAM. These algorithms are being used to process sta and user data for the Band Excitation suite [233], atom finding and local crystallography analysis [234], ptychography [235] and large spectral data sets [236, 237]. The overall workflow is shown in Figure 17.2.

The workflow process can be succinctly summarized as the following:

1. Data is generated at the "Scientific Instrument Tier" on an appropriate microscope platform.

2. The data is transferred via the "BEAM User Tier" using a local in-house connection (scp) from the microscope control resource, or a personal sta /user machine via HTTPS to the CADES resource.

3. The CADES resource a ords multi-tier architecture that simultaneously serves as a data storage repository, BEAM Web and metadata server, and the CADES Cluster Computing resource that executes parallel user workflows.

4. BEAM can then allocate jobs to additional DOE HPC platforms and allows post-authentication, "push button" execution of dynamically generated workflows.

As of now, only a small percentage (1–5%) of the data is analyzed on BEAM, due to such a short lifetime of the project. The expected use of such an infrastructure would ideally be 95% and more, with a small subset of data reserved for customized processing and algorithm development.

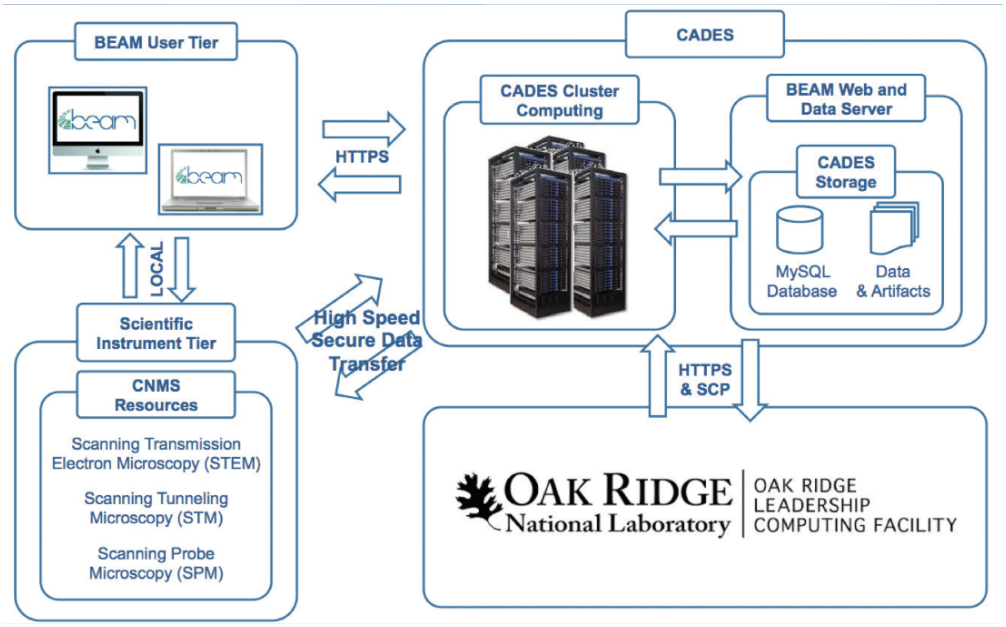


Figure 17.2: BEAM workflow and infrastructure.

17.1.2 Future Scanning Probe and Transmission Electron Microscopies

As illustrated in Tables 17.1 and 17.2 current data volumes are already approaching the capacity for analysis on a local compute resource—like a workstation computer. In the near future computational clusters will be necessary in order to handle even the simplest of operations in data visualization. It is important to note here, that unlike physical probe microscopies (AFM, Scanning Tunneling Microscopy, STM) data generation time for (S)TEM and FIB are at least *three orders of magnitude faster*. On average, a single high-quality image on a physical probe microscope is collected in approximately five minutes at 512×512 pixels; whereas in a (S)TEM, a single 4k×4k image is captured well under a second. With images being perhaps the most basic, easy to handle and process data types.

These data generation volumes extend beyond issues in processing and storage, but also in data transfer, particularly in experiments that rely on real time feedback to the tool operator. This problem is complicated even further by the fact that many of the experiments summarized may happen concurrently with parallel data flows coming from independent detectors. Combined tilt-focal series; time series spectra; or through-focus Ptychograms [238] as well as movies that are even minutes in length will take a lot of space and require massive throughputs that necessitate livestreaming capabilities from the microscopes in order to efficiently transfer this information. It is immediately apparent from the near-future trends in Figure 17.3 and Table 17.2 that these problems are only expected to get more severe.

We envision that operating within an HPC environment will provide the key interface for the intimate interaction of experiment and theory. The multimodal, hyperspectral data collected in these new generation microscopy techniques is an amalgam of what is typically independently processed by well-established, theoretical techniques that utilize self-contained approaches but are rarely cross-validated. These independent analysis workflows are well-understood, and widely utilized in a high-intensive computational environments by theoreticians today. We expect that combining storage, preprocessing, and theoretical efforts will intertwine experiment and theory into a single, streamlined analysis process enabled by an HPC environment. Naturally, the grand goal of uniting these e orts is to enable true theoretical feedback to guide experiment and discovery in near real-time.

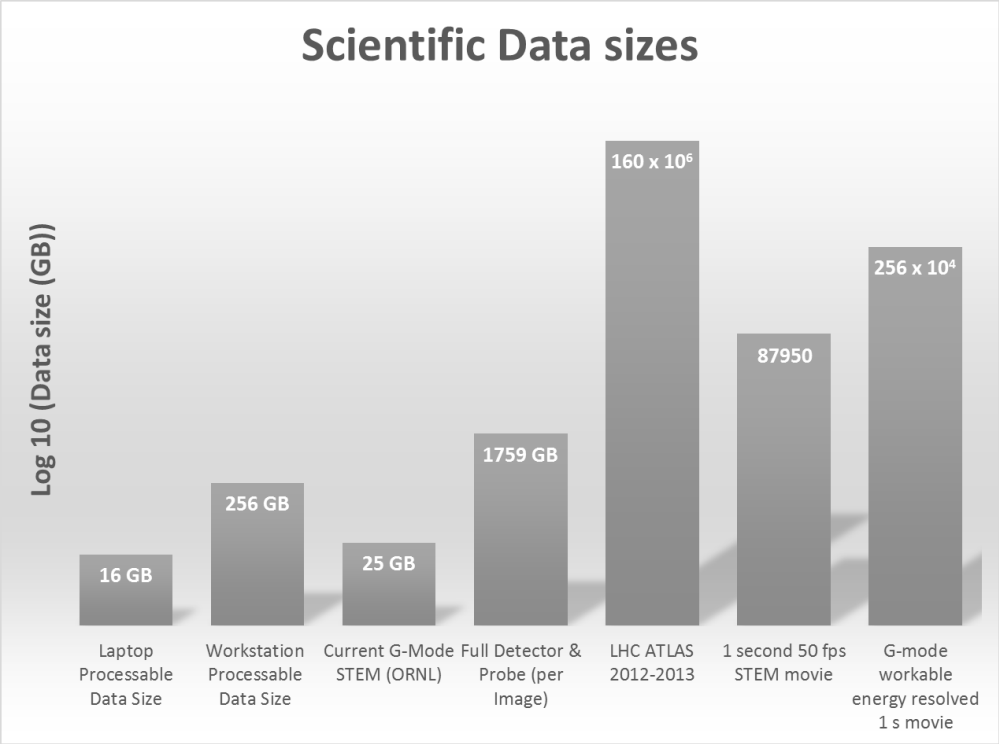


Figure 17.3: Scientific data sizes on the processing and generation ends. For columns noted from left to right, laptop and workstation capabilities are estimated by average machines available on the market today. 25 GB general-mode (G-mode) STEM is a single "small" (see Table 17.1) 4D (200×200×400×400) data set. Full detector and probe data size is for a single 4D hyperspectral data set where the output of all electron or ion probe positions (2048×2048) is captured at an *average size* detector array (2048×2048). Large Hardon Collider ATLAS detector output for 2012–2013 is also noted. A 50-frame movie captured at 2048×2048 probe positions with a 2048×2048 detector array. Modest G-mode movie captured at 200×200 probe positions on a 768×768 pixel detector (per 1 exposure), with 256 Ronchigram Energy Channels over 40 frames.

> We expect that combining storage, preprocessing, and theoretical e orts will intertwine experiment and theory into a single, streamlined analysis process; enabled by an HPC environment. Naturally, the grand goal of uniting these e orts is to enable true theoretical feedback to guide experiment and discovery in near real-time.

17.1.3 Data Lifecycle for Scanning Probe and Transmission Electron Microscopies

The data lifecycle follows a familiar cyclical pattern commonly found in Data Life Management literature which is replicated in Figure 17.4.

- *Creating Data:* The microscope generates the data almost entirely. There is some additional descriptor metadata associated with the sample, operator and the microscope state, but it is rather infinitesimal compared to the size of the detector output. The number of detectors can vary, but currently will rarely cross over into double digits. The raw data output is uncompressed and is currently at 32-bit integers (software limited), with older microscope detectors clamped at 16 bits. Data transfer mech-

Figure 17.4: Data lifecycle process in imaging.

anisms from the detector to the storage media depends on the manufacturer, but the most commonly used interfaces are USB, Ethernet and PCI/PCI-e.

- *Processing Data:* Classical processing methods utilize binning and averaging to improve the signal to noise, however over the last decade the detection hardware has improved significantly, with the processing methods largely utilized to control data volumes. At the very first stages of the analysis workflow, we are interested in collecting full detector response at the fastest meaningful rates in order to assess tool performance and adjust parameters on-the-fly. Additionally fast visualization schemes would be of use to monitor the sample and quality of the output signal (§17.1.2).
- *Analyzing Data:* The analysis framework has to be scalable, parallelizable and flexible. Due to the maturity of the field, a large number of instrumental configurations, and with the breadth of scientific interests, the analytical backend has to have the capability to adapt to either completely new analysis library software, or have the flexibility to combine analysis workflows in an unconstrained fashion. Currently the analysis requirements include standard packages for plotting, 2D or 3D visualization, file I/O, mathematical and multivariate statistical processing libraries. In the near future, more exotic neural network, image and hyperspectral registration and segmentation, compressed sensing, and robust file sharing packages will be necessary.
- *Preserving Data:* As a part of the user center service, data stewardship has to be included with the data collection and processing services. Data quotas, length of storage, redundancy, and encryption are only some of the important details that have to be discussed at the proposal preparation stages. However, some form of basic, short-term storage and access over the lifetime of the user project for pertinent analyzed and raw data sets have to be available.
- *Accessing Data:* In the current framework of the user nanocenters, the data belongs to the PI on the

user proposal. Associated students, sta , and co-PIs have access to the data with the permission of the main PI. As such we envision data access during its lifetime on the ORNL compute resources to be limited to the individuals with proper training and security credentials vetted through the ORNL system that are a part of the user project for which the data is being collected.

- *Re-using Data:* The BEAM framework at CNMS was conceived with data reusability and reanalysis in mind. Due to the nature of the experiments and the statistical framework to analyze and refine the data, recurring analysis of the same data set is vital for understanding the underlying physics and chemistry, as well as the validity of proposed theoretical models. We expect that users will reanalyze their data sets multiple times, and have in fact built the capability to capture and contain the results of periodic reanalysis into the data file structure. Additionally, the results of such a persistent approach to analysis will be cross-correlated and form some of the very basis of scientific arguments enabled by the BEAM framework.

Due to the nature of the experiments and the statistical framework to analyze and refine the data, recurring analysis of the same data set is vital for understanding the underlying physics and chemistry, as well as validity of proposed theoretical models.

17.1.4 Data-centric Requirements: Capabilities, Speeds, and Feeds

The data-generation speed in SPM is presently limited by the bandwidth of the optical detector (at 10 MHz) multiplied by the data capability of the DAQ card (16–32 bit). Typical information content in the data stream is limited by specific imaging mode, etc., but can be estimated based on typical oscillation amplitude (approximately 0.1nm in contact modes, 50–100nm in non-contact modes) and the magnitude of thermal noise in the system. Traditionally, the first step of data utilization is heterodyne filtering (lock-in or phase-locked loop) that compresses the approximate 10-megahertz data stream from the photodetector to about a 1-kilohertz data stream of amplitude/phase or frequency/amplitude data (compression is chosen to match acquisition time of single spatial pixel, which in turn is controlled by the speed of topography feedback). In band excitation mode (developed in 2007), excitation is performed at multiple frequencies, e ectively multiplexing data stream to about 100 kHz. In a recently developed G-mode (developed in 2015), the full data stream is captured. The functional imaging of the materials is achieved by scanning time, voltage, and parameter space at each spatial location, giving rise to multidimensional data sets as summarized in Table 17.1.

The acquisition of these compound data sets brings the obvious challenge of data storage, dimensionality reduction, visualization, and interpretation. While the analysis is tailored for new materials systems or detection sequences, we can summarize the typical procedure for Band Excitation (BE) Piezoresponse Force Microscopy operation. In these, the first step of data analysis of a 5D data set includes a simple harmonic oscillator fit along the frequency dimension, reducing dimensionality by 1 and giving rise to a 4D amplitude, quality factor, and resonant frequency data set (vs. position and *stimulus*). For time measurements, the data can be analyzed to yield time delay hysteresis loops (e.g., using proper relaxation function fits). Resultant 3D data sets are fitted using phenomenological models to give 2D images of polarization dynamics (and for example, their time dispersion). For first-order reversal curve (FORC) type measurements, we typically convert to the Preisach type plane and then study spatial variability of Preisach parameters. For a 5D and 4D data set, we regularly use the multivariate statistics methods such as principal component analysis (PCA) to explore the variability of materials responses and its relationship to surface morphology. The use of more complex multivariate statistical analysis tools, such as end member extraction using the Bayesian method, has also been applied to 4D and 5D data sets. Similarly, the use of independent component analysis and k-means clustering for specific problems may also be applied. Note that despite the complexity of the analysis procedure, in some cases 5D data sets can be reduced to two 2D images with readily identifiable

Technique	Dimensionality	Target data set	Target data size
Band Excitation PFM (BE-PFM)	3D, space and ω	$(256 \times 256) \times 64$	32 MB
Switching spectroscopy PFM (SS-PFM)	3D, space and voltage	$(64 \times 64) \times 128$	4 MB
Time relaxation PFM (TR-PFM)	3D, space and time	$(64 \times 64) \times 128$	4 MB
AC sweeps	4D, space, ω, voltage	$(64 \times 64) \times 64 \times 256$	512 MB
BE Polarization Switching (BEPS)	4D, space, ω, voltage	$(64 \times 64) \times 64 \times 128$	
BE thermal	4D, space, ω, temperature	$(64 \times 64) \times 64 \times 256$	512 MB
Time relaxation BE (TR-BE)	4D, space, ω, time	$(64 \times 64) \times 64 \times 64$	64 MB
First order reversal curves (FORC) BEPS	5D, space, ω, voltage, voltage	$(64 \times 64) \times 64 \times 64 \times$	2 GB
Time relaxation on sweep, BE	5D, space, ω, voltage, time	$(64 \times 64) \times 64 \times 64 \times 64$	16 GB
FORC Time BE	6D, space, ω, voltage, voltage, time	$(64 \times 64) \times 64 \times 64 \times 16 \times 64$	128 GB
FORC IV BEPS	5D, space, ω, voltage, cycle	$(64 \times 64) \times 64 \times 64 \times 16$	4 GB
FORC IV and FORC IV-Z	4D, space, voltage, cycle	$(64 \times 64) \times 64 \times 20$	200 MB
Time-resolved Kelvin Probe Force Microscopy (KPFM)	3D, space, time	$(60 \times 20) \times 10^6$	8 MB
Open loop (OL) BE KPFM	4D, space, ω, voltage	$(256 \times 256) \times 32 \times 16$	256 MB
General-mode PFM (G-PFM)	3D, space and voltage	$(256 \times 256) \times 1.6 \times 10^4$	4 GB
G-mode Voltage Spectroscopy (G-VS)	ND, Space, voltage	$(256 \times 256) \times 1.6 \times 10^6$	400 GB

Table 17.1: Development of multidimensional SPM methods at CNMS.

physical meaning (e.g., separation of reaction and di usion in electrochemical systems, or components of relaxation in a ferro-relaxor arising from field-induced phase transitions and ordinary polarization switching). However, some of the multivariate methods, such as the Bayesian unmixing approach, require HPCs to be realized on the 5D and 6D data sets, due to memory and computation requirements. General-mode (G-mode) data is typically processed in sections or chunks to alleviate the handling of large data sizes. Data sections are transformed to the frequency domain via a fast Fourier transform and multiple signal processing routines such as low-pass filters, and noise-thresholds are applied to reject noise in the signal. Alternatively, multivariate statistical analysis methods such as PCA may be applied to statistically filter the signal. Upon filtering the data, the aforementioned statistical methods are used as in BE data to extract relevant material properties.

Figure 17.5 depicts the time required to process data acquired in BE and G-mode techniques. The processing capability of a computer is generally limited by the memory and the speed, and parallel processing

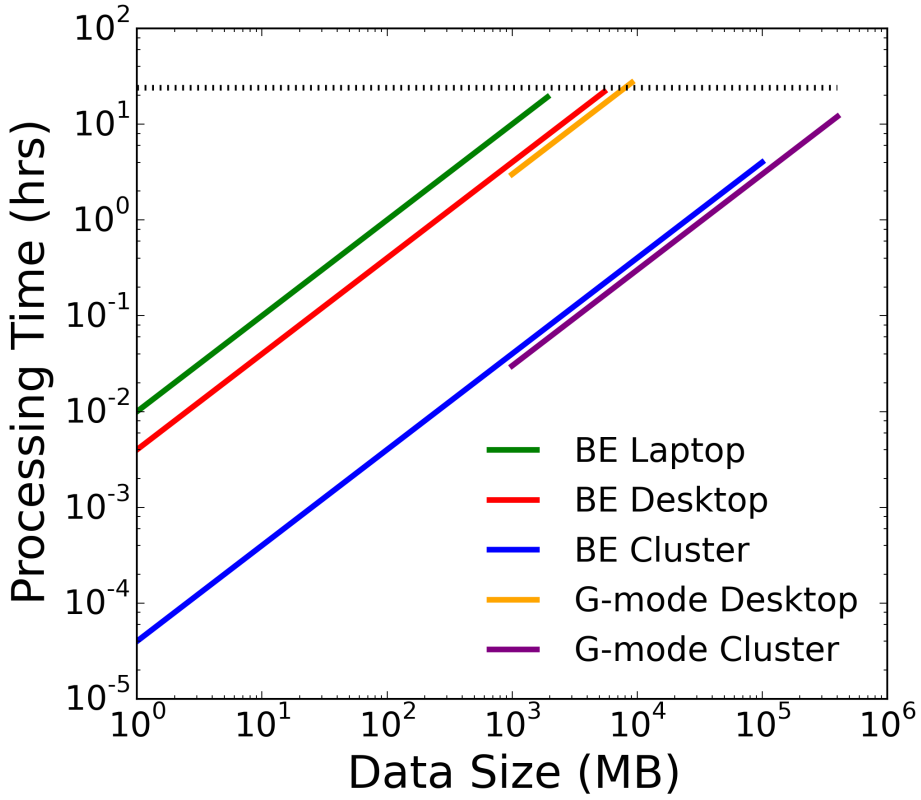


Figure 17.5: Variation in computational time for BE, G-mode data sets for a range of data sizes.

capability of the processor. Consumer laptops are feasible only for processing BE data smaller than 1 GB. Though workstations and desktops can process large BE and G-mode data sets, processing times can near 24 hours for data sets exceeding 4–10 GB. HPC clusters can provide a 100–500 times improvement in processing time and can potentially enable real-time processing.

The parameters in Table 17.1 illustrate that even for 4D methods data sampling is insufficient to ensure high temporal and energy resolution, whereas for 5D these problems are presently critical (e.g., eight FORC sets are insufficient for Preisach map sampling, and about 100 sets are typically required) and so far preclude 6D imaging, although a low resolution form has been conducted. Figure 17.6 illustrates the bottlenecks that currently limit the data generation in select experimental methods, and they can be broadly classified as limitations in:

- Data acquisition: Our current instrumentation software is capable of transferring only a limited number of data samples—10^6 to and from the data acquisition hardware. This limitation results in trade-o s between the number of FORC cycles, voltage steps in BEPS measurements, and the number of measurements that can be averaged to improve the signal-to-noise ratio. These limitations preclude larger data sizes in Tr-KPFM, BEPS than those possible currently.
- Microscope drift: Scanning probe microscopes su er from drift in the scanner position with respect to the tip. Though the drift can be neglected in measurements that span over just a few minutes (e.g., band excitation piezoresponse force microscopy, BE PFM, or general mode Piezoresponse Force

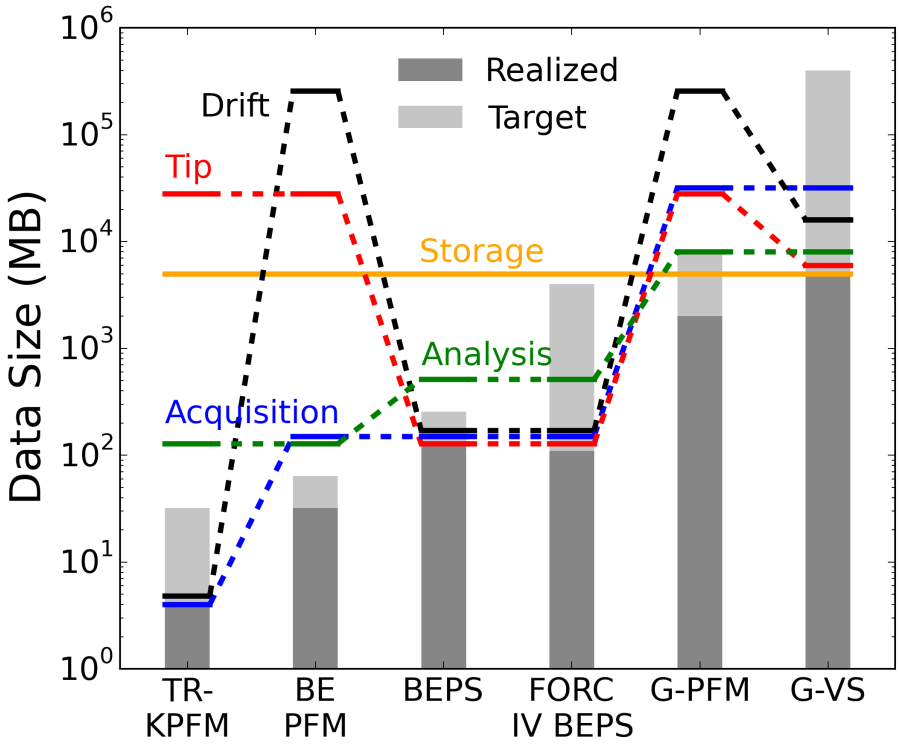


Figure 17.6: Limitations in realizing the target BE and G-mode data sizes.

Microscopy, G-PFM), the drift can be substantial in slower measurements such as time resolved Kelvin Probe Force Microscopy (Tr-KPFM), Band Excitation Piezoresponse Spectroscopy (BEPS), and FORC in current voltage measurement (FORC-IV) BEPS. The drift is exacerbated by the sudden and jerking motion of the scanner as the tip moves between measurement points. The drift forces tradeo s in the spatial, or voltage resolution in the measurements and thereby precludes the acquisition of larger data sets than those currently possible.

- Tip wear: Many techniques such as BEPS, FORC IV BEPS and G-VS apply large biases between the tip and the substrate which can result in electrochemical reactions at the tip that can erode the conductive metal layer that covers the tip. Mechanical abrasion, due to friction when scanning, can also result in tip damage which deteriorates the spatial resolution.
- Analysis: As described earlier, the analysis time scales linearly with the data size and our current techniques generate data that can require more than 24 hours of processing time with existing workstations. This limitation precludes analysis of G-mode data sets larger than 8 GB.
- Until 2014, data generated by most experimental techniques occupied only a small portion of the data storage drive and others such as FORC time BE generated larger data but at a slow rate. The newly developed G-mode techniques are capable of generating as much as 38 MB of data per information channel, every second. It is possible to fill existing storage drives with just a few such G-mode data sets in a day. Access to vast, fast, cloud-based storage drives is necessary to enable storage of such large data sets.

Table 17.2 summarizes various common experiment types and the data sizes for a given number of probe position (x, y) coordinates and higher dimensional energy channels. Looking beyond proposed values to

a five- to ten-year outlook, the data sizes will continue to grow. Realistically for each electron, we could record the probe position (x, y), scattering angle (u, v), and the energy loss (E) (here we assume that you record the energy loss as a single value, not as a complete spectrum); resulting in a 5D data set as the most basic data unit. Additionally, this data could be a function of frame or focus or some physical parameter (Ptychogram, Focal series, Tilt series, etc.) adding dimensionality and size. Recording (x, y, u, v, E) gives roughly 20 bytes of data (using 32-bit integer values) per electron, providing a data rate of 4 Gbps, for a standard imaging case of about 32pA current, which is roughly 200 electrons per microsecond per detector.

Experiment Type	Probe positions	Pixels or channels	Estimated size (32 bit int values)	Near future outlook (32 bit int values)
Spectrum (Note 1)	1	1k	4kB	-
Ronchigram **ref 26**	1	1k × 1k	4 MB	16MB
Line spectrum	128	1k	512kB	16MB
Image	1k × 1k	1	4MB	64MB (4k × 4k) × channels
Spectrum image	64 × 64	1k	16MB	16GB (1k × 1k × 4k)
Ptychogram **ref 25**	64 × 64	256 × 256	1GB	4TB (1k × 1k × 1k × 1k)
Focal series	512 × 512	160	167MB	1GB (2k × 2k × 160)
Tilt series	1k × 1k	100	400MB	1.6GB (4k × 4k × 100)
Time series	512 × 512	100	100MB/frame	Many 4k × 4k × 100 frames (hours)

Table 17.2: Current trends and short-term outlook for data generation and sizes in electron and ion beam microscopies. Note 1: Spectra could be Electron Energy Loss Spectra, X-ray spectra or other detector feed. Note 2: These are intended to be "typical" values based on the ORNL systems that people are currently using, rather than the maximum value possible. Depending on the sample, set-up or particular microscope hardware, values could easily increase by a factor of 2–4.

Processing stage	Present/Near-term	Long-term
Data acquisition rate: maximum rate(s) and monthly or annual totals	~10–100 GB/day for movies, ptychographic data sets, and Gmode SPM	~10 Mb/s for SPM, ~(1-10) GB/s for STEM in the full information capture modes
Experiment-side processing	Lossless compression, exploratory data analysis/multivariate statistics, deconvolution, feature extraction, pan-sharpening, compressed sensing, image registration	data reduction, metadata collection, collaborative analysis, real time theory feedback
Real-time constraints, turnaround time from collection to result for experimental control	In most cases offline in the day-month interval for analytics, minutes-hours for microscope operation	Real-time analytics (unmixing, atom finding, structure extraction) at imaging rates
Metadata/provenance capture	Metadata from instrument and environmental parameters. Storage of data analysis pathways	Capture appropriate analysis meta data and notebook style information. Cross-correlation of metadata with literature/web/data base searches

Table 17.3: Data and analysis requirements in scanning probe and electron microscopies

17.1.5 Impediments, Gaps, Needs, Challenges

Regarding Data Capture, Compression, and Storage

1. Real-time processing and high-data throughput to support real or near-real-time experimental feedback is a challenge with data transfer rates of at least 4GB/s required.
2. We will also need large, accessible data repositories for archival and data sharing, with annual storage capacity of 5 PB.
3. Limited access to sufficient computing resource for the initial processing of generated data is another challenge with a requirement for a dedicated 2000 (64 per microscopy tool) node system with GPU capabilities in the near future.

> . . . what hinders the future of STEM/FIB can be summarized as the *lack of resources to move, store, share and process scientific data*.

While each of the aforementioned roadblocks is a challenging issue at even the current rates of data generation and analysis, what hinders the future of STEM or FIB can be summarized as the *lack of resources to move, store, share and process scientific data*. According to even the modest near-future estimates outlined in Table 17.1 and Table 17.2, continuous, secure data transfer rates approximately 4 GB/s *per microscope* are required to sustain tool operation to adequately fulfill the center's obligation to the user community. An alternative but temporary solution is efficient data compression at the generation point. Furthermore, global data accessibility is an important preamble to analysis—as input from various experts in the field is vital to achieve real scientific progress. Therefore, a flexible data repository that allows fast and secure access to data by a handful on individuals to advise and oversee the analysis process is critical. Finally, the ability to receive real-time, or near-real-time feedback to the operator requires a dedicated computational resource to each microscope. In the near future, satisfying both data processing and theory requirements for the data stream o each microscope is likely unrealistic, however previewing the analyzed data and getting it ready for serious theoretical e ort is an excellent near future goal.

Data analytics and visualization

- Data visualization for high dimensional data sets (note that given highly regular spatial grids, its likely to be less complex then for more abstract data sets)
- Infrastructure to support, verify, and reuse custom codes for mapping to physical models (e.g. recognoton and classification based neural nets algorithms, Bayesian endmember extractions, and other component analysis methods).
- Image registration (for mapping multiple data set over time, e.g. aging of battery or upon changing gas pressure or global temperature when consecutive images can be shifted spatially and must be aligned e.g. using topography as a reference)
- Development of physics-based statistical tools (e.g. constrained unmixings, etc.).

Workforce training

Many of the data curation problems and the extraction of materials-specific responses from instrument data require close interaction between domain scientist and data scientist. Generally, such dual backgrounds are

an exception and some amount of professional training (boot camps or intensive courses) is required to fill this gap.

> Many of the data curation problems and extraction of materials specific responses from instrument data require close interaction between domain scientist and data scientist. Generally, such dual background is exception and some amount of professional training (boot camps, intensive courses) is required to fill this gap.

Infrastructure for data re-use and integration across user facilities

The development of universal framework for full data capture within individual facilities further brings forward the consideration of their integration, providing an integrated data environment for subsequent re-use and data mining. This necessitates discussion of the integrability between chosen architectures and data formats.

Conclusion

In conclusion, the emerging trends place a heavy emphasis on combinatorial imaging that correlates spatial, chemical and physical information. Serious challenges in processing these data have been slowly, but steadily addressed by the scientific community on many fronts, however scaling, validating and cross-correlating these independent e orts is a serious roadblock that can only be addressed by close collaboration with the HPC and data handling experts. We foresee that close ties with the information technology community would usher in a technical revolution in the scientific fronts by providing the infrastructure for truly close-knit multidiscipline collaboration.

Case Study 18

Computing within the Advanced Photon Source for Data Collection and Analysis

Brian H. Toby
Argonne National Laboratory

18.1 Background

The APS[1] is the Nation's premier high-energy light source. Nearly 70 distinct beamlines are operated, with approximately half run by the facility and the rest by externally run consortia. Each beamline operates with unique capabilities and an independent scientific mission. Thus, it is better to think of the APS as a confederation of more than 70 di erent independent laboratories (since some beamlines support multiple scientific missions) rather than a monolithic experimental facility. Computational needs and strategies may di er considerably across beamlines, but computation is required for nearly every aspect of the facility. At present, it is estimated that the APS generates *circa* 2 PB of raw data/year. This document will concentrate on computing within the X-ray Science Division (XSD) of the APS. An assessment of XSD data and computing needs are underway, and a strategic plan for APS scientific computing is also currently being prepared.

Each beamline operates with unique capabilities and independent scientific mission. ... Computational needs and strategies may di er considerably across beamlines, but computation is required for nearly every aspect of the facility.

The APS is planning to upgrade to a multi-bend achromat lattice storage ring to go into operation in the early 2020s. This will increase the brightness of the APS by two to three orders of magnitude and also the coherence of the source significantly. While not all beamlines will gain equally from the upgrade, some of the APS beamlines that already have the highest data rates can be expected to be enhanced by one to three orders of magnitude.

[1] Information about the APS can be found at: http://aps.anl.gov.

18.2 Science Use Case

While a comprehensive consideration of computing challenges for the entire facility is impossible without a very extensive document, it is possible to consider a single scientific domain, as is done in the following section.

18.2.1 Example: X-ray Imaging/Microscopy Challenges

Scanning probe microscopy and X-ray imaging have made tremendous impacts on the scientific community over the past decades, addressing extremely broad and highly relevant scientific questions. In the life sciences, X-ray fluorescence microscopy (XFM) has had a revolutionary impact in the area of bioinorganic chemistry, directly enabling the visualization of trace metal content, with various applications studying the role of metals in life and disease (fundamental biology, cancer research, endogenous dysregulation of metal homeostasis, development of therapeutic drugs and contrast agents, nanomedicine, bioremediation, metal impact on carbon cycling in the ocean). In the energy sciences, XFM has been used to study impurities and contaminants in photovoltaic materials to understand and improve upon the limitations in device performance, or to study geopolymers as a low carbon alternative to Portland cement. Bragg di raction with a focused beam as a local structural probe is one of the most recent techniques to transition to nanoscale microscopy, and can probe local structure in a crystal, orientation, morphology, and both elastic and plastic strain in single crystal, polycrystalline, composite, or deformed materials in 3D. The ability to easily alternate between polychromatic and tunable monochromatic di raction modes enables studies of a wide range of randomly oriented or polycrystalline, "real" materials—this makes possible a direct and quantitative comparison of real samples to theory and simulation on the mesoscopic length scales of crystalline materials.

A key limitation today is our ability to analyze and visualize the acquired data due to its volume, velocity and variety. For example, consider the example of mapping trace elemental content in a zebrafish embryo. [239]

> A key limitation today is our [in]ability to analyze and visualize the acquired data due to its [increasing] volume, velocity and variety.

This particular data set (see Figure 18.1) was acquired at a lateral spatial resolution of 2 μm and only 60 projections. A significant challenge in its reconstruction was the very low signal levels, which led to the use of (slow and computationally expensive) iterative reconstruction methods. Current methods rely on first geometrically aligning individual projections (because measurement geometries are uncertain), and then carrying out reconstructions. Therefore, the quality of the end result very heavily depends on sufficient signal statistics to align individual projections. Also consider the potential gains o ered by techniques such as dose fractionation, [240, 241] which demonstrate that 3D data sets could be acquired in essentially the same amount of time as a 2D data set, as long as individual projections can be aligned. The spatial resolution in a reconstructed tomographic data set depends on both the lateral resolution of the microscope employed for its acquisition as well as the size of the sample and the number of projections used to image it. The implication is that to achieve the highest spatial resolution for a data set as the one mentioned above one wants to both acquire at the highest resolution as well as over many more projections. The resolving power of the above data set was limited to about 5μm, due to signal levels. With the upgraded APS and improvements in detectors, we will be able to acquire the data set at a 200nm spatial resolution, but without advances in data analysis methods, due to poor signal-to-noise levels it is impossible to align and reconstruct the model parameters for data comprised of roughly 4200 projections having 4200x7810 datum each. Assuming conventional data acquisition approaches where one might acquire full X-ray fluorescence

spectra, one would have at each scan point 4 spectra with 2048 values each, assuming 4 bytes per channel, this data set corresponds to about 4 petabytes worth of data. In order to fully exploit the principle such as dose fractionation and to reconstruct the highest fidelity 3D tomographic data sets that are solely limited by radiation damage, we need to develop and implement approaches that are able to combine reconstruction and alignment into a single step, and use all the available information (e.g., numerous di erent X-ray fluorescence channels) simultaneously.

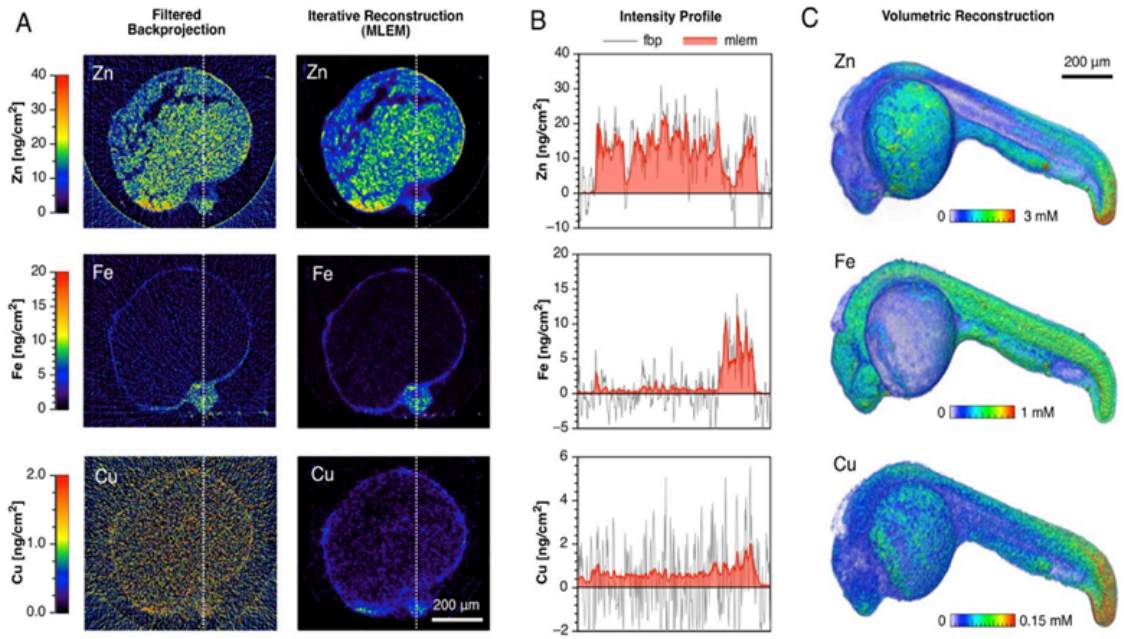


Figure 18.1: Visualization of the elemental distribution in a zebrafish embryo by X-ray fluorescence tomography (MLEM reconstruction). (A) 3D rendering of the embryo indicating the spatial orientation of the virtual slices that are displayed in panels (B) and (C). Slices include a sagittal and transverse section (top), a coronal section (middle), and a sagittal section o set to the left (bottom). (B) Elemental distributions of Zinc (Zn), Iron (Fe), and Copper (Cu) for each of the 4 slices. Individual concentration scales for each element are displayed at the bottom of each column. (C) False-color overlays of the elemental distributions of Zn, Fe, and Cu indicating regions of colocalization. The concentration scales of each element were normalized and color-coded as follows: Zn (green), Fe (blue), and Cu (red). Areas of colocalization appear in the corresponding mixed hues. Reprinted from Bourassa, et al., Metallomics 6(9): 1648-1655. [239]. The significantly improved performance of advanced approaches such as MLEM is directly visible in the comparison with more standard approaches such as filtered-back projection, in particular when looking at noisy data such as the Cu signal. Even then, significant further improvements could be achieved by using all the available information in the reconstruction and inverting the data set as a whole as opposed to first aligning individual projections and then carrying out a reconstruction piece by piece. Today we are missing corresponding software tools able to carry out these tasks.

In addition, recent progress in coherent lensless imaging clearly points to the ability to combine one such approach (ptychography) routinely with scanning probe microscopies to extend spatial resolution for the imaging of structures well beyond the limitations set by optics. [242, 243] While significant improvements have been made in the parallelization of reconstruction methods, [244] a key challenge will be to apply these methods to the simultaneous inversion of a full 3D data set, in particular when acquired using approaches such as dose fraction as outlined above. For example, consider the sample *Chlamydomonas reihnahrdtii* reconstructed in Deng, J., *et al*. [242] where a 2D ptychographic data set was reconstructed from

more than 25,000 individual coherent di raction patterns each recorded in images with about 100,000 pixels each. Assuming a four-byte precision per pixel, the data set corresponds to about 9 GB in size. A 3D tomographic data set of this kind would constitute about 1 TB. Again simultaneous 3D reconstruction is required to provide best results, and ultimately enable 3D spatial resolution only limited by radiation damage on the sample, but we are far from able to address these problems with today's software and approaches.

Lastly, some experimental errors in data acquisition (e.g., uncertainties in object or detection geometry) are unavoidable, hence the need for post-collection alignment of projections, or better models of the object that can capture these dynamical e ects. With sufficiently advanced algorithms it is possible to attempt to correct these placement errors through appropriate modeling, in the actual reconstruction of the data set, since one has multiple measurements of the same sample, however it becomes a very large and difficult problem.

Advancements in area detector technology allow collecting full-size (2048×2048) images at kHz and faster speeds. At this unprecedented rate, the integration of fast continuous (fly) scans observed via complementary metal-oxide semiconductor (CMOS) detectors and the high X-ray flux available at synchrotron facilities, allow for tomography of dynamic systems, i.e. to collect multiple full-size tomographic data sets per second and to generate 3D movies of evolving samples. When collecting tomographic data of fast evolving samples, the data collection is set at the highest speed required to capture the transient phenomena of interest that is still compatible with a sample and its environment. In these circumstances, to prevent undesired sample shrinking or movement during data collection, the instrument operator usually reduces the number of projections and the exposure time to reduce the sample motion artifacts, at the cost of angular undersampling and reduced detector signal-to-noise. The choices of scan parameter selection is often based on the operator experience. Very short exposure time and noisy data emphasize the detector non-linearity, scintillator defects and beam motion, leading to extreme artifact and making the 3D reconstruction and segmentation very challenging.

> Predicting the optimal scanning parameters, such as the detector exposure time, number and optimal angular position of the projections could optimize data collection schemes and ultimately provide better quality data. ... Besides predicting the optimal scanning parameters, the analysis of the resulting data then becomes the next bottleneck preventing near-real-time error detection or experiment steering.

Predicting the optimal scanning parameters, such as detector exposure time, number and optimal angular position of the projections could optimize data collection schemes and ultimately provide better quality data. This approach can lead to novel X-ray tomographic technique with improved temporal resolution by more than an order of magnitude compared to conventional data collection schemes and has been of great interest to a wide range of communities, looking at an out-of-equilibrium pattern forming system, like growth morphology of metallic dendrites (see Figure 18.2) and battery failure processes and material characterization since the material morphology sets the properties of many metallic alloys. [245]

Besides predicting the optimal scanning parameters, the analysis of the resulting data then becomes the next bottleneck preventing near-real-time error detection or experiment steering. Rapid tomographic image reconstruction via large-scale parallelization is a new approach that leverages highly parallel computers to improve the performance of iterative tomographic image reconstruction applications. These methods have been applied to the conventional per-slice parallelization approach, but these are limiting the number of cores to the number of sinograms (typically 2048). A recently developed in-slice parallelization approach, [246] can use many more processors and has been demonstrated to reduce the total reconstruction times for large data sets by more than 95% on 32000 cores relative to 1000 cores. Moreover, the average reconstruction times are improved from 2 hours (256 cores) to 1 minute (32000 cores), thus enabling near-real-time use. We still need multi-level optimization approaches to further improve the accuracy and

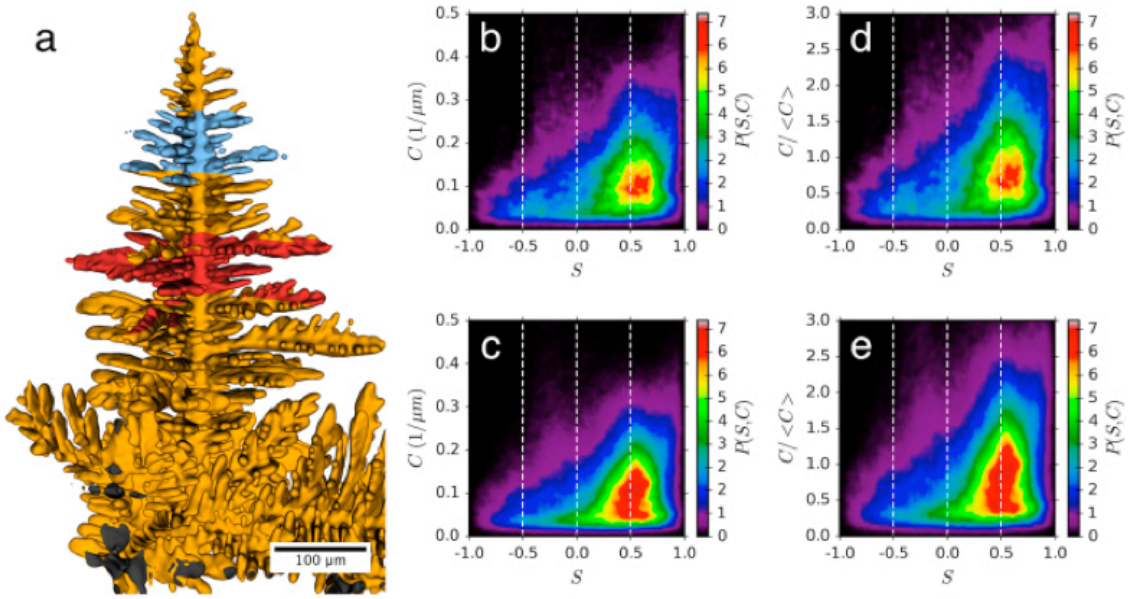


Figure 18.2: Interfacial shape distributions (ISD) for two 75μm thick slices normal to the growth direction of the nearly free-growing dendrite at 9.0 seconds after nucleation. The blue section is centered at 125μm from the dendrite tip and corresponds to the ISDs in b and d. The red section is centered at 250μm from the tip and corresponds to the ISDs c and e. The ISDs in d and e have their vertical axes scaled by the average curvature.

accelerate solution of the problems.

18.2.2 Facility Overview: Computing in Near Term

XSD computing is performed as part of our mission to enable research using X-rays and develop scientific methodologies utilizing X-rays [247]. As part of that, computation may be employed for any or all of the following processes, depending on the beamline in question:

- to conduct the experimental measurements,
- to verify that beamline operation is progressing properly,
- to reduce raw data to a form for optimal interpretation,
- to analyze raw or reduced data, including the fitting of models to observations,
- to simulate expected experimental results from a model, and
- to catalog raw, reduced and/or transformed data sets and provide them to users.

Most of the software utilized for the above is developed within the APS. While most experimental research is done by external facility users, XSD scientists often initially develop X-ray science data analysis methodologies. XSD scientists also perform their own research utilizing APS beamlines and may also collaborate with users on data analysis. The software for this comes from a multitude of sources, but a large amount is developed by APS beamline sta for their own use. This software is sometimes supported at a level that allows it to become more widely used.[2] In recent years, professional software engineers have also developed

[2]Scientific software packages developed at the APS are documented online at `https://www1.aps.anl.gov/Science/Scientific-Software/`.

or reworked a few scientific software packages for use at APS beamlines. In coming years, it is expected that teams consisting of computational engineers and scientists together with beamline scientists will have a greater role in this process.

Experimental data reduction and analysis computation is usually done either on beamline workstations, which can be micro-clusters, or on a 53-node cluster run centrally by the APS. Nodes on this cluster are configured for and dedicated to specific APS beamlines, so that hardware is always available on demand. While some beamlines have piloted use-cases on leadership-scale computing for data reduction and analysis [248, 249, 246], at present these facilities are not utilized on a regular basis for any part of APS operations. We plan to migrate some of our offline tomography reconstruction computations to Mira (ALCF) in the next year, but most beamlines require computational resources within minutes or perhaps seconds when data are available and cannot abide with the queued structure employed on leadership machines. Argonne's Laboratory Computing Resource Center plans to prototype a virtualized on-demand compute service in the coming year, which we hope can be useful for beamline operations.

. . . in the next year, but most beamlines require computational resources within minutes or perhaps seconds when data are available and cannot abide with the queued structure employed on leadership machines.

With the throughput demanded by the pace of experiments and the requirements for scientists to manually perform workflow tasks, beamline sta are left with little time to directly work with users on data analysis and interpretation tasks. How much data is "left on the table" can only be guessed upon. It certainly depends on the field and the beamline. Certainly, as greater automation of experiments and data analysis is implemented, sta time is freed for a greater involvement in science, which can only improve facility productivity.

The biggest challenge to the facility is how to create the scientific software needed to run it: software for improving the experimental process; for implementing beamline data movement and reduction workflows; to perform preliminary quality assurance, visualization and reduction; for data analysis and interpretation; for automating analysis workflows and distribution to users.

The biggest challenge to the facility is how to create the scientific software needed to run it: software for improving the experimental process; for implementing beamline data movement and reduction workflows; to perform preliminary quality assurance, visualization and reduction; for data analysis and interpretation; for automating analysis workflows and distribution to users. The process of software development has been eased in part through languages such as Python, which has a high adoption rate amongst experimental scientists, but the process of adapting working algorithms to utilize continually new parallel computing architectures has only become a more demanding process. It is not enough to simply create software. Packages will never see their full potential without user outreach, written guides, and worked-through examples/tutorials. As soon as maintenance and development of a package ceases, *rigor mortis* will soon set in. Thus, software projects carry with them a mortgage for as long as they will be utilized. Further, each APS beamline has its own unique portfolio of data reduction and analysis packages, so the total number of codes needed is quite large.

A lesser, but still pressing problem relates to handling the ever-increasing volumes of data generated by beamlines due to advancing detection technologies. As one example, we expect a technique that is already a high-data volume generator, X-ray photon correlation spectroscopy (XPCS), to see an order of magnitude increase in data rates in the *coming months* as a new detector is delivered. A still faster detector is already on the market and even faster detectors for XPCS are being developed. At present, XPCS data are processed in

near-real time, but it will be a significant challenge to continue to move these data and process them at that pace as the volume increases. For XPCS, the reduced data are much smaller than the raw images and there is little need to provide the latter to users. This is di erent for many other techniques, where large data sets are provided to users. For example, in tomography the resulting tomogram is of comparable size to the raw images, which are also usually retained so that di erent reconstruction algorithms can be compared.

Every APS beamline can identify significant software needs, enough to keep at least one professional busy on an ongoing basis; in addition there is a significant backlog of unfulfilled facility-wide needs. However, not even in the most optimistic budget scenarios will staffing for scientific computation development at the PAS approach a level of even 0.5 FTE per beamline. Clearly, only part of the challenges can be met, but work can be most e ective through collaboration, and by implementing integration that allows for greater code reuse. The development of tool kits that speed code development also need to be a part of this strategy.

Every APS beamline can identify significant software needs, enough to keep at least one professional busy on an ongoing basis; in addition there is a significant backlog of unfulfilled facility-wide needs. . . . Clearly, only part of the challenges can be met, but work can be most e ective through collaboration, and by implementing integration that allows for greater code reuse.

Beamline computing falls into four categories. (A) On-the-spot computations to examine measurements as they are collected; (B) after-the-fact transformation and data reduction; (C) post-experiment data interpretation; (D) simulation and modeling based on experimental findings. This last case is most readily accessible for leadership-scale computing, for example with molecular dynamics (MD) or Density Functional Theory (DFT) calculations to simulate model systems, but this is not a typical use case and is also not a facility focus. The other types of beamline computing cannot be discussed without considering scheduling demands. There is little point in predicting yesterday's weather. Likewise, beamlines need to provide feedback to experimenters as to how their measurement is working while the experiment is in progress and entering such computations into a queue means that measurements may be conducted "in the dark."[3] Only a small fraction of computing can tolerate delays on the scale of hours, and even then beamline scientists will still choose the computing strategy that provides the fastest throughput and requires the least e ort.

To date, computing user facilities have not been utilized to any significant level in routine APS operations. The key to changing this is in establishing mechanisms that allow beamline computing to preempt the long-running batch jobs that provide the main demand for these large machines.

To date, computing user facilities have not been utilized to any significant level in routine APS operations. The key to changing this is in establishing mechanisms that allow beamline computing to preempt the long-running batch jobs that provide the main demand for these large machines. It should be noted that the most e ective beamline computing will be scaled to use the largest amount of resources that can be deployed e ectively in order to provide a result to a user within minutes if not seconds of completion of the measurement. This means that the ideal use cases (when parallelization is possible) will employ large numbers of processors for short periods, with potentially long delays between tasks as the next set of data are collected. Thus, by design, these computations will optimally use only a small fraction of the facility's capacity, making shared use a high priority.

[3] See https://www.alcf.anl.gov/articles/boosting-beamline-performance for an example of how spotting a trivial problem early can spell the di erence between a successful outcome and a wasted week of beam time and associated travel costs.

18.2.3 Facility Overview: Future

In the near future, we expect to see significant expansion of data rates due to improvements in detector technology, which is progressing at a rate far outstripping Moore's law. An even larger dislocation is anticipated due to the APS upgrade, which will allow a growth in data rates for some instruments by factors of 10^2 to 10^3. The APS upgrade will permit multiple techniques to be applied to a single sample concurrently. Analysis of such multimodal techniques will produce extremely complex data streams, which will require complex reconstruction techniques to be "knit together" and sophisticated analysis techniques to help users find meaning from multidimensional data sets that are too large and complex for humans to mentally integrate. The assessment report that is nearing completion will review the computing needs for each APS beamline as well as sizes of current and anticipated data streams.

Simulations of experiments, in advance of measurements, are not commonly performed. Facilitating this process will improve facility e ectiveness in two ways: (A) Proposal reviewers can judge if an instrument has the required sensitivity to perform the intended experiment; (B) users can come to the beamline prepared, knowing what aspects of the measurement will best di erentiate between anticipated models.

18.2.4 Data Lifecycle

At present, most raw data are collected at beamlines as collections of related images, often on computers dedicated to detector operations with limited software capabilities; additional information may be distributed across multiple other computers and databases. Images may be collected in bursts as fast as 2000 single-megabyte image frames per second, but several orders of magnitude improvement will be expected in the coming 5–10 years. The data are typically reduced and in some cases only the reduced data are retained, but in other cases the raw images are kept. The data of record are provided to users who have the responsibility for archiving them, as determined by the data management plan associated with the user's funding program. When data sets are large, data are typically provided to users on removable hard disk drives, but an increasing number of users are employing Globus Transfer [15, 16][4] to move data over networks.

One particular area of interest is in workflow tools. These are of need for beamline task integration/data management and for data analysis pipelines. A related need is for HPC scripting, for which the Swift language [250][5] has proven quite useful for certain classes of problems. Workflow programming represents a common need across all science user facilities, and resources devoted to a tool that satisfies a large cross-section of beamline needs would be welcomed by the APS.

The APS is currently improving data pipelines to ease the e ort required to transfer data between storage facilities, including staging for computing within the ALCF and to a facility-run Globus endpoint. At present, the APS does not provide a centralized and robust long-term data archive, as this is categorized as a user responsibility. The facility may be called upon to provide this as a service in the future.

[4] http://www.globus.org.
[5] http://swift-lang.org.

18.3 Impediments, Gaps, Needs, Challenges

> The most serious impediment the APS encounters is a lack of a DOE-wide view of software needs across the BES mission. Since each lab has its own portfolio of responsibilities, it devotes resources to those goals. For software development, in some cases the most e ective mechanism for accomplishing them is for facilities to join together with coordinated programs.

The most serious impediment the APS encounters is a lack of a DOE-wide view of software needs across the BES mission. Since each lab has its own portfolio of responsibilities, it devotes resources to those goals. For software development, in some cases the most e ective mechanism for accomplishing them is for facilities to join together with coordinated programs. In many cases, due to lack of staffing with appropriate specialized knowledge or facility prioritization, it makes more sense to have a team at a single laboratory develop a package that suits needs of all of BES. Two very e ective examples of this from Argonne have been TomoPy [251] and GSAS/EXPGUI [252], which are both used world-wide. In fact, EXPGUI is at present cited approximately 500 times/year, but it is also no longer being supported due to staffing requirements. As budget pressure on labs becomes more severe, software development from facility operational funds by necessity must be more focused on internal priorities. *DOE-wide needs will not be properly met unless experimental data analysis software development projects are resourced outside of facility operation and unless projects have development metrics encompassing needs of all facilities.*

> DOE-wide needs will not be properly met unless experimental data analysis software development projects are resourced outside of facility operation and unless projects have development metrics encompassing needs of all facilities.

There is a tremendous potential synergy for ASCR to collaborate with BES science user facilities for the gain of both communities. In fact, as discussed previously, the APS will not be able to make full use of future multimodal types of data without considerable involvement from computer scientists. However, creating manageable and highly productive e ort profiles for such cooperative projects is a tricky problem to manage. Involvement from scientists from both communities is required. Without participation from beamline scientists from the targeted facility(s), there is a high likelihood of a carefully designed tool that no one wants will be produced. A successful project will have clear deliverables that include a working, though likely incomplete, version of the software as a deliverable within the *first third* of the intended project duration. A functionally complete version of the code is needed in the *second third of the project*, allowing the remainder of the time for documentation, user-demanded improvements and user outreach. Regular workshops are needed with members of the intended user community. In the early stages, these will be design and user interface reviews. Towards the end they become educational outreach. The plan must also address how the software will evolve over an extended duration and how maintenance and outreach will continue. As soon as this ends, *rigor mortis* sets in; to build software without a plan that encompasses the intended lifetime of use is akin to demolishing a building as soon as construction is complete.

> Coordinated prioritization and production of core tools that would aid software development across multiple science user facility organizations would be a very wise investment for the user facilities.

The software development process is greatly benefited by the ability to utilize libraries and programming environments. When programmers utilize common standards and toolkits, it also encourages interoper-

ability between packages. Examples of tools that speed the development of code include Doxygen and Sphinx, which build developer's documentation from source code, NumPy and SciPy, which provide a scientific computation toolkit, and HDF5, a robust hierarchical file format. However, when it comes to development of HPC code, there are fewer tools that ease the process for scientist-software developers (as opposed to computational experts) to transition from prototype code to HPC production code. Given that Python is a particularly welcome environment for scientific code development, improving the process of porting from Python to exascale will greatly aid the utilization of leadership-level computing for X-ray science. Coordinated prioritization and production of core tools that would aid software development across multiple science user facility organizations would be a very wise investment for the user facilities. High-quality extensible workflow development tools appropriate for workstations through HPC deployment are also very much in need. Likewise, well-developed and supported data standards would very much benefit facilities. At least at our facility, the NeXus format has failed to achieve much penetration due to lack of adequate software toolkits and basic tools. The most powerful driver for adoption of a standard is a "killer app" associated with the standard that entices users to demand compatibility throughout the scientific food chain. Tools that allow domain scientists without HPC skills to easily develop parallelizable code in a programming environment they prefer, is at present most commonly Python. Many scientists are very comfortable translating their data analysis concepts to computer code, but are most comfortable and productive doing so in an interpreted language environment like Python, Matlab or R, but not in Java or C++. At present, significant e orts from HPC experts are needed to adapt such codes to make e ective use of even the multi-core processors found in laptops.

As discussed before, for beamline computing to share hardware with the traditional use of leadership computing requires that beamline needs preempt long-running tasks for short periods. This requires new approaches for scheduling and requires rapid task startup and switching. Without on-demand access, most beamline computing will not be deployed at SC user facilities; without preemptive scheduling, such use will not be e ective or welcome. Ideally, this use case would also be combined with containerized or virtualized computing, as this eases the configuration process, which can become complex as the number of packages in use at a facility grows.

New approaches to data set management and storage are needed. Conceptually, one thinks of data as a single or perhaps a hierarchical set of files in a single place. In practice, at our facility a data set is accumulated as a set of files and sometimes database entries that are dispersed amongst multiple computers, as they are collected. After the experiment and during data reduction and analysis, these files are migrated to other locations and additional intermediate results are added. Further, if one considers the frequent occurrence that multiple researchers work on analysis with periodic restarts and overlapping approaches, a data set looks more like a distributed github project than a hierarchical directory. Another welcome addition in the data universe would be a centralized DOE facility that provides a Globus-integrated mechanism for data archival and retrieval, that could be provided as an option to users at cost. The Petrel system at ANL[6] is used for this purpose at present.

In conclusion, the APS has many unmet needs, but the ones of greatest concern are now to create, support and port software to use HPC facilities.

[6] http://petrel.alcf.anl.gov.

Case Study 19

Data Challenges in the Deep Underground Neutrino Experiment

Tom Junk[1], Amir Farbin[2], Maxim Potekhin[3], Craig Tull[4], Xin Qian[3], Brett Viren[3] and Chao Zhang[3]

[1] Fermi National Accelerator Laboratory
[2] The University of Texas Arlington
[3] Brookhaven National Laboratory
[4] Lawrence Berkeley National Laboratory

19.1 Science Use Case

19.1.1 The Deep Underground Neutrino Experiment

The Deep Underground Neutrino Experiment (DUNE) will provide a unique, world-leading program for the exploration of key questions at the forefront of particle physics and astrophysics. Chief among its potential discoveries is that of matter-antimatter symmetry violation in neutrino flavor mixing. Other exciting physics objectives include the possible detection of supernova bursts and the search for nucleon decay. DUNE is the successor of the Long-Baseline Neutrino Experiment, which is documented in detail in "Oxygen octahedron reconstruction in the srtio(3)/laalo(3) heterointerfaces investigated using aberration-corrected ultra-high-resolution transmission electron microscopy" [253].

DUNE has been conceived around three central components:

- An intense, wide-band neutrino beam (700kW upgradeable to 2.3MW),
- A fine-grained Near Neutrino Detector (NND) just downstream of the neutrino source, and
- A "Far Detector" based on a massive, 40000 ton Liquid Argon Time-Projection Chamber (LArTPC) deep underground, but still 1300 km downstream from the source of neutrinos. There will be approximately 1.5 million channels in the device. The TPC will also incorporate an integrated photon detector.

The DUNE Far Detector will consist of four individual LArTPC modules 100000 ton each. A conceptual diagram of these modules placed in the cavern is shown in Figure 19.1.

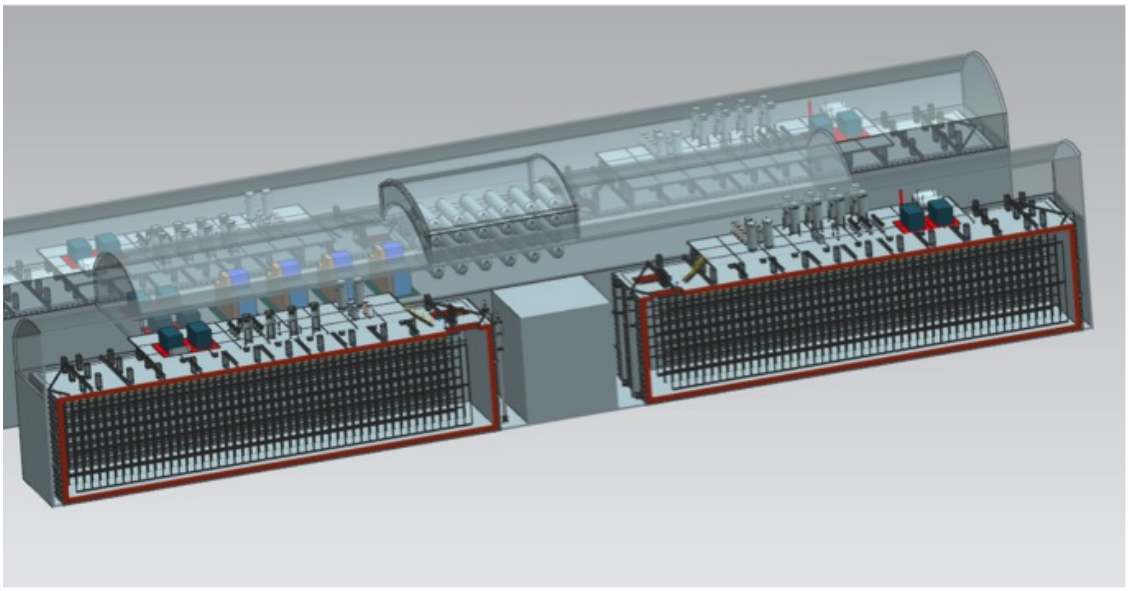


Figure 19.1: DUNE Far Detector schematic.

The LArTPC volume is instrumented with wires spaced at about a 5mm pitch, which are attached to rectangular supporting frames to form "induction" and "collection" planes. Each collection plane is accompanied by two induction planes configured as a stereo pair at a certain angle (§19.1.2). The planes are aggregated into units called "Anode Plane Assemblies," or APA. The wires are read out as individual channels, with waveforms recorded at a two-megahertz digitization frequency. A set of parameters of the DUNE detector relevant to characteristics of its data stream is listed in Table 19.1.

These parameters create a unique large-scale and highly granular detector, but also bring about an array of challenges due to characteristics of the data being produced. For example, if one was to record *all* the data coming from the detector, corresponding data sets would be in hundreds of exabytes worth of data annually. However, since the detector is well shielded from cosmic rays, most of the data will be due to noise. Of specific importance is the decay of ^{39}Ar (see §19.1.5) naturally present in Argon in trace quantities, which is a beta emitter with the endpoint of 565 keV. However, most of such signals can be rejected by applying "zero suppression" (ZS) techniques whereby portions of the digitized waveform which are consistently below a certain threshold are discarded.

While ZS does reduce the rate and amount of data significantly, there are science objectives in DUNE such as the detection of supernova bursts (SNB) which require continuous recording of data at low thresholds for a considerable amount of time. This requires the careful design of the online farm which is needed to implement corresponding logic, data bu ering and transmission strategies, and thorough Monte Carlo studies to optimize the system.

19.1.2 Near-Term View

In the near term, DUNE must address a few related but distinct work areas which require computing at scale. Most important of these are covered below, followed by examples of dataflow in DUNE.

Parameter	Value
Full height (module)	12.0m
Full width (module)	14.5m
Full length (module)	58.0m
# of detector modules	4
channels per APA	2,560
APAs per module	150
Active height (APA)	6.0m
Active width (APA)	2.3m
Drift distance in Liquid Argon	3.6m
Drift velocity	1.6mm/μs
Drift time	2.25ms
bytes/sample	1.5
sample rate	2.0MHz
# drifts/readout	2.4
readout time	5.4ms
samples/readout	10,800
Total # of channels	1,536,000

Table 19.1: Fundamental parameters of DUNE Far Detector LArTPC.

LArTPC prototypes

The 35t prototype

There is a LArTPC prototype built at FNAL, with capacity of 35t of Liquid Argon (hence commonly referred to as "the 35t prototype"). It will start taking data with cosmic rays in early 2016. In addition to validation of a few engineering solutions, it will generate data which will help to test elements of reconstruction algorithms for LArTPC.

protoDUNE

The name "protoDUNE" was given to the full-scale single-phase LArTPC prototype to be deployed at CERN in 2017 for measurements with a test beam provided by a special target and purpose-built beamline from CERN Super Proton Synchrotron (SPS). It will serve to validate various DUNE technology aspects before proceeding with the construction of the principal DUNE detector, and will also provide an important platform for realistic LArTPC detector characterization (e.g., PID, shower response, etc.) utilizing controlled conditions of a test-beam experimental setup.

It is foreseen that the total amount of data to be produced in protoDUNE will be of the order of 1PB (including commissioning runs with cosmic rays). Processing these data and conducting Monte Carlo studies connected to the experiment will require substantial resources and planning. The protoDUNE experiment will not be shielded like DUNE (to be built in a deep cavern) and thus will be subject to substantial occupancy from the cosmic ray muons, resulting in background conditions and data characteristics quite di erent from those of the eventual full detector.

Reconstruction methods R&D

Time Projection Chambers used in experiments like STAR at Relativistic Heavy Ion Collider (RHIC) and A Large Ion Collider Experiment (ALICE) at the LHC [254] feature pad-based readout scheme, which allows for a relatively straightforward reconstruction of 2D patterns in a given time slice. However, using pads in

larger detectors is not practical due to power consumption considerations (and requisite cooling requirements), excessive cost of readout electronics and other factors. For this reason, due to its sheer size, the DUNE LArTPC features wire-based readout to cover its extremely large fiducial volume while keeping the channel count realistic. This makes its large scale possible, but also leads to a loss of spatial information being available for reconstruction (when compared to pads). This creates challenges for disambiguation of ionization signal loci and therefore for event reconstruction. Reliability and thorough characterization of the algorithms employed in this area will be critical for the systematics and other performance characteristics of DUNE.

There are a few approaches currently in development for event reconstruction. The "Pandora" toolkit, which originated as R&D for fine-grained calorimetry at ILC [255], is being adopted to reconstruct LArTPC events. In addition, there is a "projection matching algorithm" which will be used for studies with LArTPC prototypes.

There is also a promising toolkit under development (called "Wire Cell") based on a di erent approach. It performs three-dimensional imaging of events using the principles commonly applied in tomography. As is frequently the case in tomographic reconstruction with sparse data, this may require the use memory- and CPU-intensive computing platforms.

It is very likely that meeting the demands of such calculations will require adopting emerging technologies that are now becoming more common in HEP, such as GPU or other co-processor acceleration and/or massively parallel systems such as HPC facilities. It will be important for this software to be ready in time for the protoDUNE operation, as described in the above item. More details on this are presented in Section 19.1.2.

Near detector R&D

The Near Detector is a significant component critical to the overall physics performance of DUNE. It will combine elements of fine-grained tracking and calorimetry and will be placed underground at FNAL. It is currently in initial stages of R&D, which will certainly require substantial Monte Carlo studies and appropriate computing resources.

DUNE distributed computing: an outlook

At present, DUNE depends on facilities provided by FNAL for the bulk of its computing power. FNAL is also expected to host a full replica of data recorded by protoDUNE. As a result of protoDUNE processing needs that will materialize within the next two years, as well as a new CPU-intensive reconstruction algorithms (such as "Wire Cell") that will be applied to protoDUNE and other DUNE data, this architecture is likely to shift in the direction of a more distributed network of data centers. There are plans to keep full or partial replicas of protoDUNE data at NERSC and BNL in addition to the "master copy" at FNAL, and to use distributed HPC resources on the Grid, including those available on an opportunistic basis.

Distributed computing in DUNE will also incorporate HPC facilities to cover specific needs of those parts of workflows which benefit the most from application of these technologies. Plans for the HPC component are now in a preliminary stage since the DUNE software makes use of such a capability is now being developed and will take some time to mature. Contacts have been established with the Computational Science Initiative (CSI) at BNL in order leverage relevant expertise at BNL and to explore options regarding HPC resources that may become available through that venue.

In parallel to establishing all the elements of infrastructure necessary for protoDUNE, there will be vigorous science tools and R&D programs in the near- to medium-term, such as the near detector, beam and

target optimization, photon detector etc. Success of these tasks will depend on the availability of adequate computing resources. In addition, the engagement of multiple institutions and researchers who are members of DUNE will be facilitated (and their resources better leveraged) if the collaboration manages to maintain portable and accessible software that can be used at any particular institution, and run transparently on modern Grid and cloud resources.

Hierarchy of dataflows in DUNE

DUNE has to deal with dataflows at a few di erent levels of scale and complexity. For example, high-bandwidth transmission and extremely fast (but relatively basic) transformations of data are taking place within the DAQ and its associated computing farms. Portions of the data coming out of the DAQ undergo quick processing to render monitoring information crucial for the experiment diagnostics and operations, thus there is a distinct monitoring dataflow. Then, there is the distribution of raw data to mass storage facilities and processing centers where it undergoes multiple transformations which typically fall into the "production" category and involve the application of calibration data (which are themselves subject to a separate and potentially complex dataflow), and reconstruction of event features such as tracks, energies, particle identities etc. The data thus derived is fed to yet another chain of transformations usually described as "analysis."

Any of these "macro" dataflows typically include the manipulation of data elements, their dependencies and transformations on a smaller scale (but sometimes perhaps with greater complexity), such as during the reconstruction of a single event or searching for an event in a TPC readout window. In the following, examples of both types of dataflows are presented.

Data flow example 1: protoDUNE

As one example of dataflows relevant for DUNE in near-term, the flow of data in protoDUNE experiment is schematically represented in the diagram in Figure 19.2. The diagram reflects only the experimental data as it progresses from being recorded in the data acquisition system, to mass storage, distribution across participating sites, processing and reconstruction, and does not include Monte Carlo and other studies done in preparation for the experiment.

It is worth noting that there are several similarities between protoDUNE and the DUNE experiment as a whole, such as

- Remote location of the detector with regards to the primary processing center (FNAL)
- Replication of data to multiple storage locations
- Multiple sites involved in processing the data, and a federation of storage across sites
- Likely inclusion of HPC in the technology portfolio

The top third of the diagram (labeled "A") depicts data undergoing transformation and movement on site at CERN, and this configuration is conceptually similar to what is currently used in ATLAS and CMS, although at a smaller scale and with less complexity. After the readout via DAQ and the subsequent application of Zero Suppression (ZS), the data is then distributed to a few bu er nodes at the same location as the DAQ to ensure sufficient bandwidth to disk and redundancy to prevent outages. There will be a dedicated network connection to CERN Central Services of about 10Gbps in order to ensure adequate headroom in data transmission and allow experimentation with zero-suppression thresholds and other online parameters which could result in rates higher than nominal.

CERN EOS (a high performance distributed storage system) serves as the next destination for the data, from which it is committed to tape (CASTOR at CERN) and also replicated to FNAL (as a full copy) and

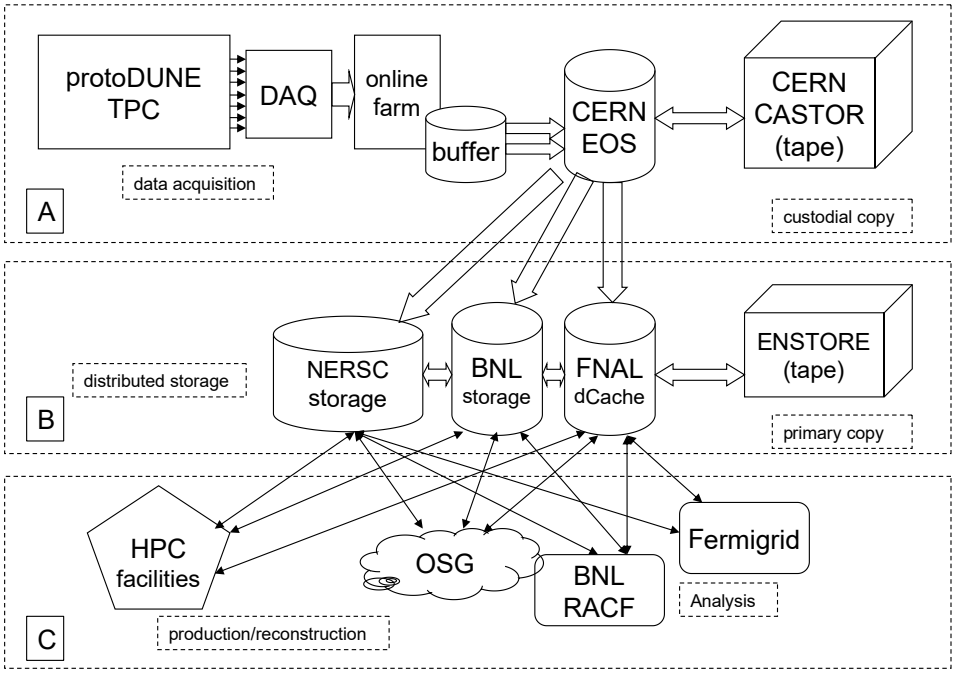


Figure 19.2: Data flow in protoDUNE.

to a few auxiliary sites such as NERSC and BNL (section "B" of the diagram). This will be done reusing tools previously developed for other experiments and facilities (e.g., IFDH/SAM maintained by FNAL or "Spade" used in the Daya Bay Experiment). The middle tier of the diagram represents distributed and permanent (tape) storage facilities in the United States, which will serve the production and analysis need of protoDUNE.

Finally, as shown on the bottom section "C" of the diagram, computing workflows will then be deployed on the resources available which will include "traditional" high-throughput systems such as Fermigrid or resources federated through the Open Science Grid (OSG), and also HPC facilities. In the analysis stage, data is expected to be shared through a federation built upon XRootD.

Data flow example 2: Wire Cell

The Liquid Argon Time Projection Chamber

In order to properly describe the problem of event reconstruction in DUNE, it is helpful to first introduce a few facts about operation of the apparatus. We start with a diagram in Figure 19.3, the meaning of which will be detailed in the following.

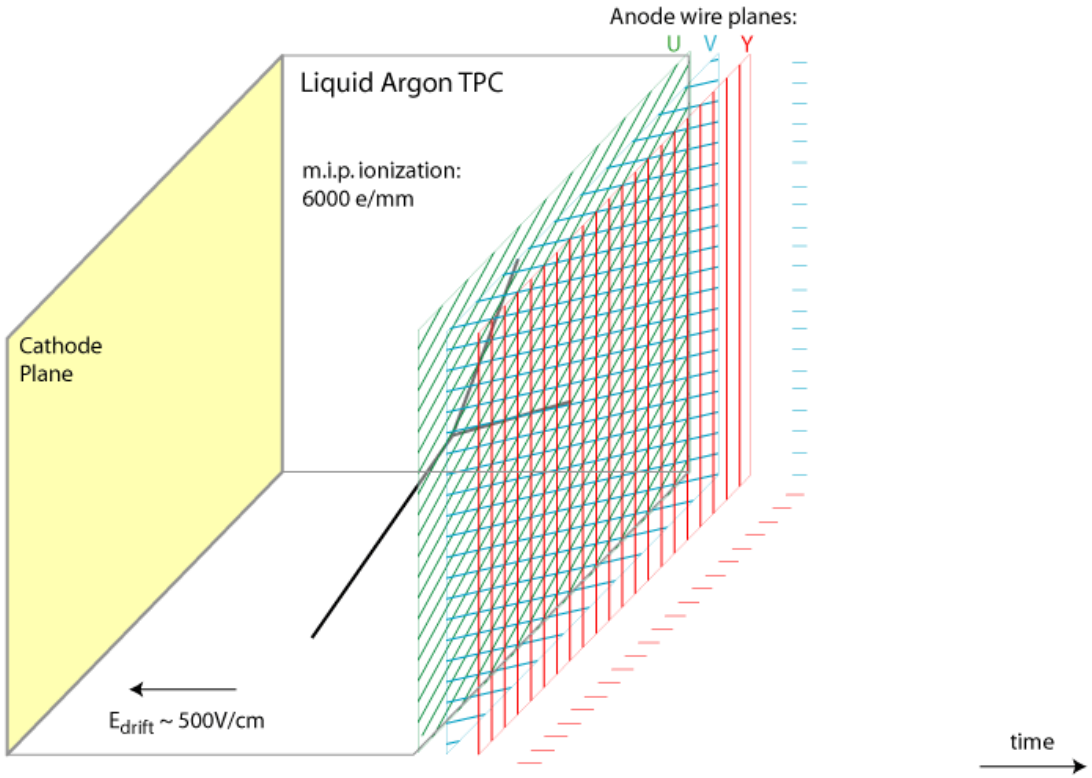


Figure 19.3: Liquid Argon TPC with wire readout: principle of operation.

The Liquid Argon Time Projection Chamber (LArTPC) is in essence a specially instrumented ionization chamber. Charges (electrons and positive ions) created due to passage of ionizing particles through the sensitive medium (argon in this case) are subject to the e ect of a uniform electrostatic field which is created in Liquid Argon by a system of cathode and anode electrodes, which causes them to move (drift) along the field lines. If there is an additional electrode within the Liquid Argon volume in the vicinity of the drifting charge, there will be a signal induced on it. Multiple such electrodes (sensors) provide means for spatial characterization of the ionization charge distribution in the sensitive volume (which for example may correspond to a particle track). Importantly, the shape of the signals on the electrode vs. time is used to measure charge localization along the drift direction (hence the term "Time Projection Chamber"). For example, ionization electrons which are closer to the collection electrode will arrive to it sooner than more distant ones, therefore time evolution of the signals on the a ected wires will reflect distribution of the charge along the drift axis.

As mentioned in Section 19.1.2, current design of large-scale LArTPC devices features planar arrays of wire electrodes supported by frames. Such design contains an essential element called *Anode Plane Assembly* (APA), which includes the "collection plane" (anode) and two planes of sensor wires, called "induction planes," oriented at stereo angles with respect to each other and the collection plane. Due to stereo angles, such an arrangement allows for the 2D measurement of the charge density distribution in the APA plane. This is illustrated in Figure 19.3, which schematically shows the drift volume (to the left), the induction planes "U" and "V" and the collection plane "Y." An important feature of such arrangement is that the *same drifting charge* is measured three times as it is detected by the three wire planes. This is further illustrated in Figure 19.4 as a schematic of drifting charge creating signals on wires, represented conceptually as a view along the direction of the drift.

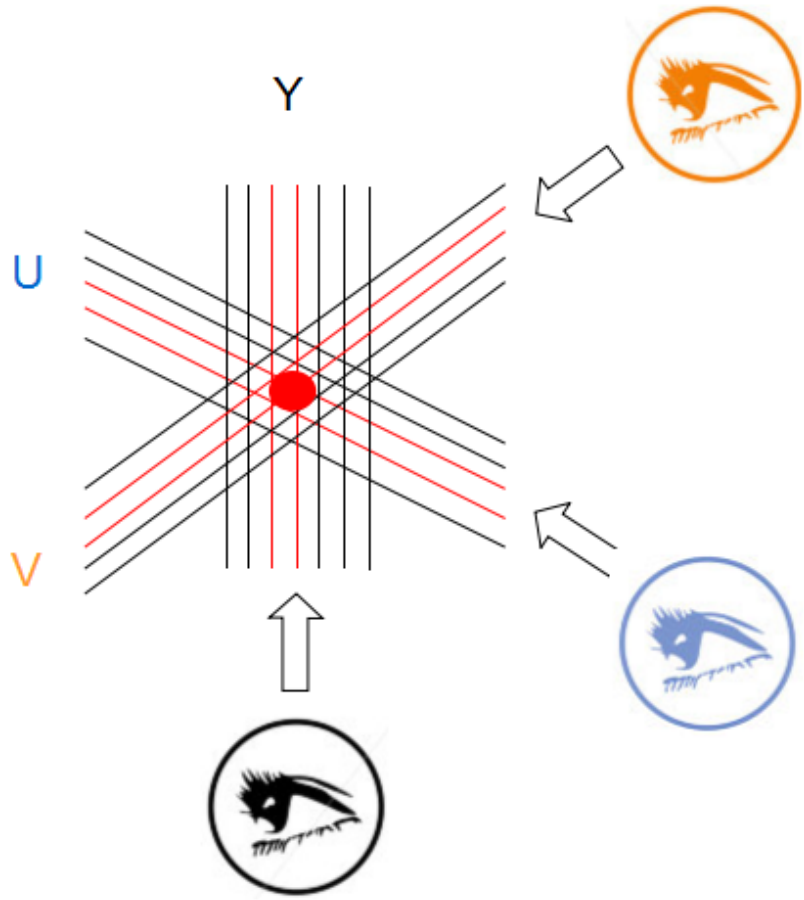


Figure 19.4: Three projections of an object in a TPC with wire-based readout.

The Inverse Problem

The LArTPC acts an imaging device, i.e. the physics information about the processes taking place in the detector volume is extracted by analyzing the 3D structures (images) of ionization patterns produced by particles participating in these processes. The only source of information available for the reconstruction of these images are amplitudes of signals coming from the wires in (U,V,Y) recorded as a function of time. It follows that the event reconstruction problem in DUNE TPC is a fairly typical case of the *Inverse Problem*, where a 3D structure must be calculated based on a set of observables.

Because of time quantization inherent in the operation of analog-to-digital conversion, the 3D image effectively becomes an assembly of 2D slices. In a given time slice, the 2D charge density distribution is observed via three di erent projections along the axes (U,V,Y) (see Figure 19.4). There is significant similarity between this type of inverse problem and Computed Tomography (CT) with limited projection data. This similarity becomes even more prominent as we observe that in a given slice the charge signals on wires are essentially linear integrals of the 2D charge density along each of the three observation axes. Common with many tomographic applications, the reconstruction strategy then consists of calculating patterns in each 2D slice and then combining them into a full 3D structure. There are many event types and topologies in DUNE, one example of a simulated neutral-current event in presented in Figure 19.5.

It is easy to see that (again, similar to the majority of tomographic applications) the reconstruction problem in DUNE is an ill-posed one, due to the very limited set of observation angles (three). At the most basic

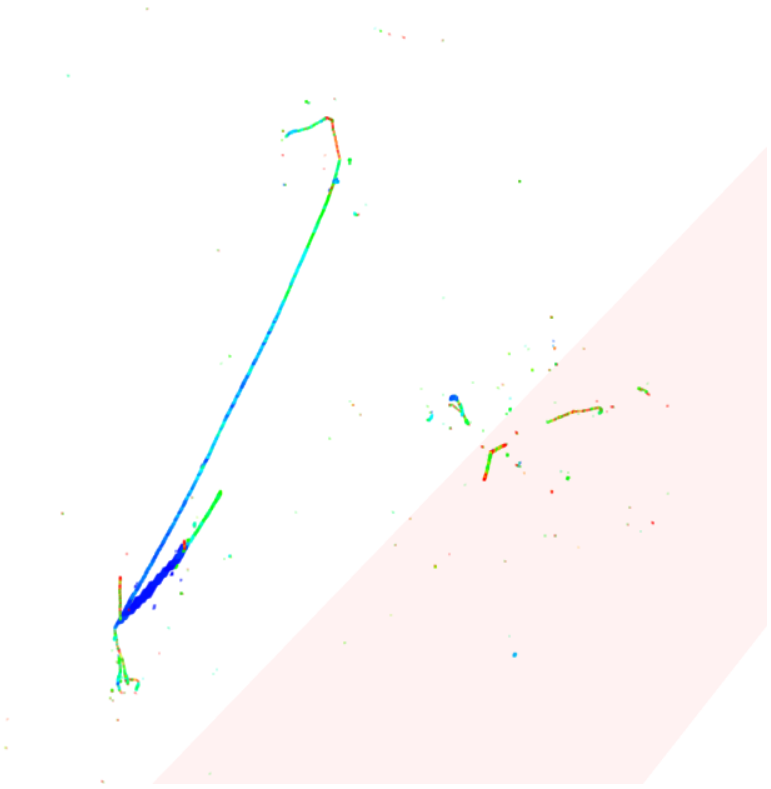


Figure 19.5: An example of a neutrino-induced reaction in Liquid Argon (simulation)

level, this issue manifests itself as "ghost hits" (well known in HEP), which is an ambiguity inherent in stereo projection measurements.

The "Wire Cell" approach to the problem is based on the following:

- "Voxelization" of the TPC volume by treating each 2D slice as a tessellation (i.e., consisting of polygon-shaped tiles);
- Solving an optimization problem which maximizes the likelihood of a given configuration of tiles (with charge associated with them) producing the observed signal distribution on the wires; and
- In reference to the optimization problem described above: regularization based on testing hypotheses about the object topology, e.g. that it is a track in a given portion of the volume.

To make this approach possible, there is one prerequisite that must be met, and that is a precise measurement of charge on each wire. This involves proper calibration of the detector as well as a solution of yet another inverse problem—deconvolution of the detector and electronics response while reconstructing the original shape of the charge signal. This is done in conditions of non-zero noise and involves the application of digital filtering techniques.

Flow of data in Wire Cell

The flow of data in Wire Cell is presented in Figure 19.6.

For the elements of the dataflow for the Wire Cell, which are significant or present most computational challenges are as follows:

Slicer: takes one "time frame" or a "readout" of raw data from the DAQ (or simulated source) and separates

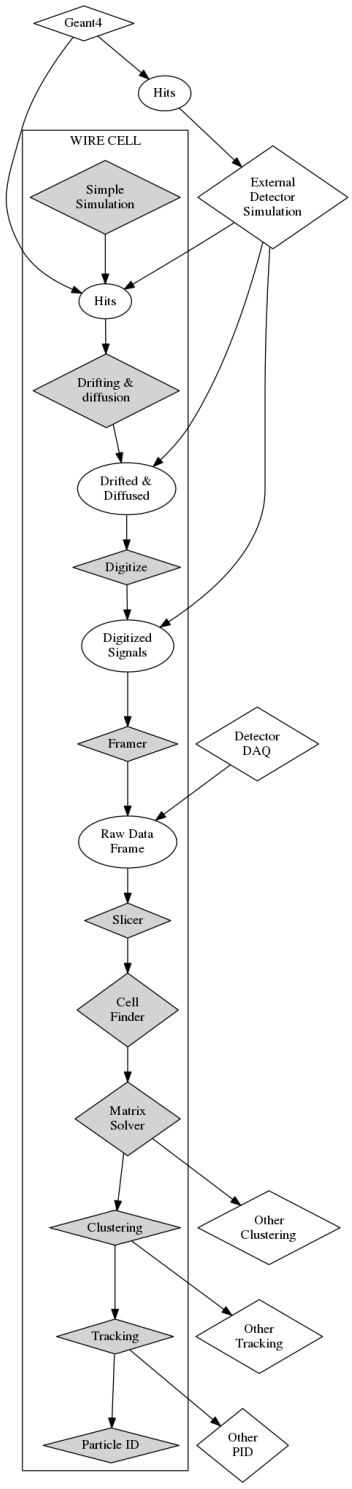


Figure 19.6: Dataflow in Wire Cell.

it in time bins. Each bin contains ADC values for all the channels (i.e., wires) with signal above a

certain threshold that span the time bin.

Cell Finder: takes data as a time bin and identifies "cells" (groups of adjacent tiles).

Optimization Solver (a.k.a. a matrix solver): optimizes the likelihood of the charge contained in contiguous groups of cells with respect to producing the observed signals on wires, utilizing a model where this relationship is expressed as a matrix.

Clustering: aggregates groups of cells over multiple time bins, thus creating 3D objects ("clusters").

Tracking: connects clusters according to certain geometry rules, forming track candidates.

PID: determines the type of particle based on the ionization pattern, as defined by charge depositions along the particle trajectory.

Elements listed above are computationally complex for a number of di erent reasons. For example, the *Cell Finder* performs a search on an array of tiles and given the large number of those may face a combinatorial challenge. The *Matrix Solver* has to solve an optimization problem involving a sparse matrix. The *Tracking* component has again to find the relations between multiple pieces of data based on the applications of certain rules.

Visualization

Wire Cell has a companion visualization toolkit called *BEE*, which provides an interactive graphic representation of various elements of Wire Cell reconstruction process to the user. An example is given in Figure 19.7, depicting a 3GeV ν_e which interacts in argon and produces a vertex with an outgoing 190MeV electron (going downward in this display) and a 559 MeV K^+, (long hooked track) and also a proton and π^- track. The activity away from the vertex is a K-long decay. The *BEE* is Web-based, receives data from a server and renders it using WebGL.

19.1.3 Future

The DUNE experiment will be commissioned in the mid-2020s, and will utilize expertise and lessons learned from protoDUNE. There will be a significant evolution of software and computing components as the experiment nears its commissioning milestones. Some of the more important changes are:

- Flexible and well-characterized methods of the DAQ and online data reduction, and compression will be put in place (some of this work is being done currently but more improvements are expected in the following years). This functionality will be deployed in the DAQ system and its online farm.
- Data rates and volume of data expected in DUNE will depend on what is attainable in the online systems, which will have natural limitations due to space, power and cooling requirements which are at a premium given its location deep underground at the Sanford facility. This item is now a subject of R&D and not all the answers are known. Even though some estimates for DUNE indicate that it will nominally generate tens of petabytes of data annually, it is all but certain that capabilities of DAQ developed in the run up to the experiment will be sufficient to reduce this volume by an order of magnitude (see comments in §19.1.5).
- By the time DUNE is getting ready to record experimental data, it will be important to have in place a fully distributed computing infrastructure which would facilitate access to data and software physics tools for all members of the collaboration.
- Reconstruction methods currently being develop are meant to be automatic, and this is achievable in many but not all cases due to intrinsic limitations of the wire-based design of the detector which sometimes creates ambiguities in mapping wire signals to anticipated event features. This can be mitigated by computer-assisted visualization analysis tools, where an operator would use his or her

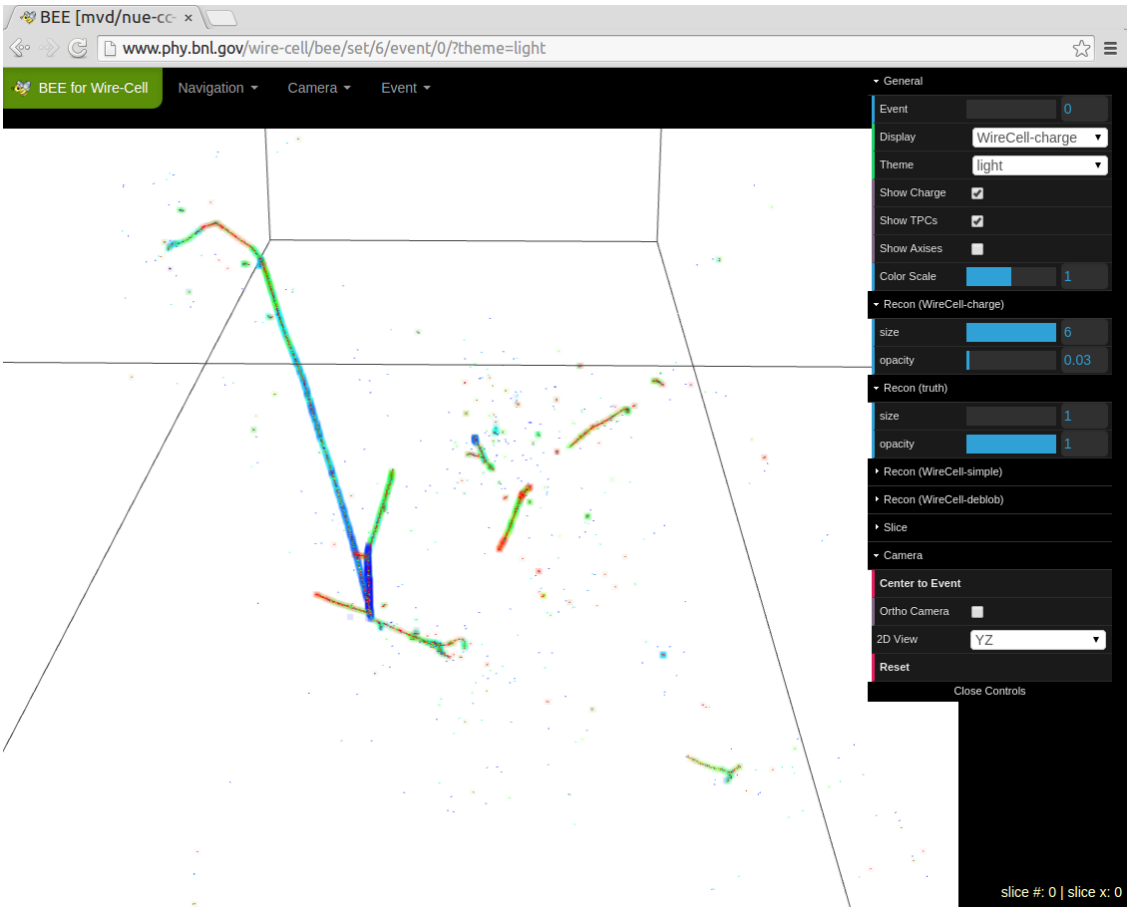


Figure 19.7: Event display in Wire Cell visualization component ("BEE"): a 3GeV ν_e interaction in argon.

pattern recognition capabilities based on the reduced, processed and 3D-rendered data coming from the reconstruction chain.

Figure 19.8 presents a simplified dataflow diagram for the DUNE Far Detector. Apart from the obvious di erence in scale with regards to protoDUNE, there will be other di erences such as:

- Adequate bu er-type storage at 'the 'Far Site" according to current plans, but no tape (section "A" of the diagram) storage.
- The number of sites to which data is distributed will be larger than just two or three, to ensure ease and speed of access to the data by most research centers of the Collaboration (section "B").
- By the time of its commissioning, DUNE will have a functional Workload Management System (WMS), based on a federation of Grid or cloud sites (section"C").

19.1.4 Data Lifecycle

DUNE incorporates a few subsystems, such as the Near Detector, Photon Detector built into the Far Detector TPC and a number of others. For the sake of brevity, only the LArTPC of the Far Detector will be considered here from the dataflow point of view, since it is by far the largest source of raw data in DUNE and likely presents most challenges for data handling.

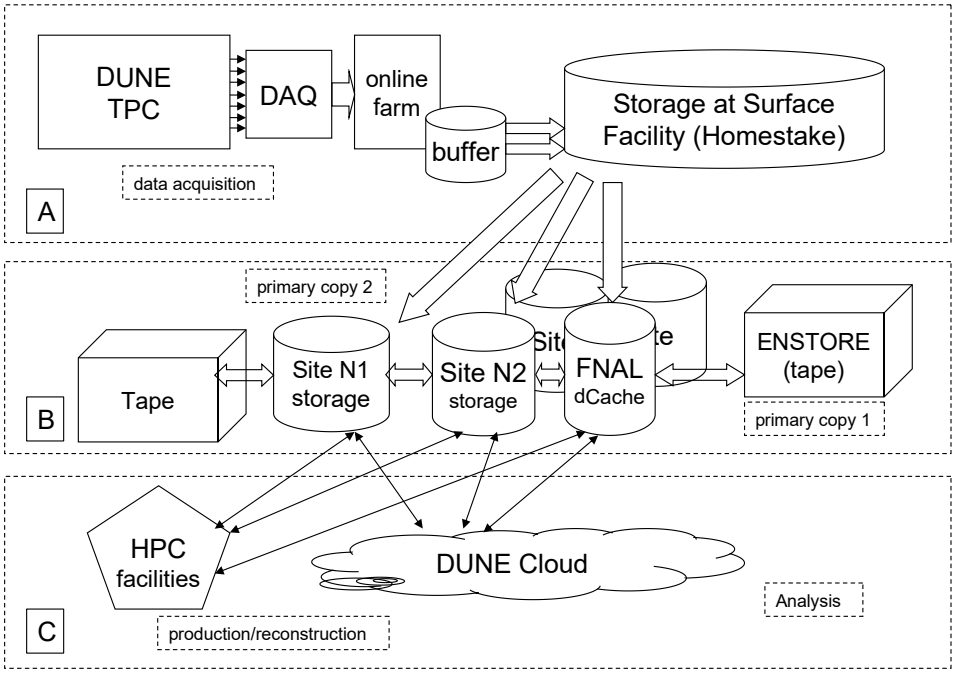


Figure 19.8: Dataflow in DUNE (concept).

The DUNE data lifecycle starts with amplified signals collected from individual wires being digitized by ADCs at about a 2MHz frequency. Within the DAQ, the data is processed in two parallel streams—the "trigger" stream and the "data" stream, whereby the processors in the data stream are reading out only those portions of data in the short-term bu er which were considered of interest according to algorithms running on the processor's the trigger stream. In order to deal with unusual event signatures such as supernova candidates (when the whole volume of the TPC "lights up" with clusters of moderate energy dispersed throughout the volume), there is also a ring bu er which keeps data long enough for the trigger farm to run corresponding algorithms and retrieve the data in case there is a suspected positive.

Next, there are three basic design elements in the data transmission and storage chain, motivated by the need to preserve data which is precious due to the high cost of operating both the facility at FNAL and the detectors that are part of DUNE. These elements are:

Bu ering in the cavern at (4850 feet below ground level) for the DAQ systems to mitigate possible downtime or outage of the network connection to the surface facility, and also at the surface facility to mitigate the downtime of the network connection between the Far Site and FNAL.

Robust transmission —data transfer needs to be instrumented with redundant checks (such as checksum calculation), monitoring, error correction and retry logic.

Redundant replicas are a common practice in industry and research (*cf.* the LHC experiments) to have a total of three copies of precious data, which are geographically distributed. Such geographical distribution of the replicas may include countries other than the United States, where the data will be collected. This provides protection against catastrophic events (such as natural disasters) at any given data center participating in this scheme, and facilitates rebuilding ("healing") lost data should such an event occurs.

Once the data is transmitted from the Far Site to FNAL and committed to mass storage (tape), additional replicas are created at other selected DUNE sites for redundancy and optionally, to allow for the wider distribution of the production workload over participating facilities. The XRootD storage federation will be deployed to facilitate the following elements of the dataflow:

- Improved efficiency of Grid jobs by providing an on-demand source of data in addition to centrally managed and more static data distribution;
- Better shared access to data in the analysis stage.

It is anticipated that similarly to HEP experiments final stages of analysis will be done using compact sets of processed data which are easily accessible over the network using interactive user facilities and computing devices (laptops etc).

19.1.5 Data-centric Requirements: Capabilities, Speeds, and Feeds

Specific DUNE use cases were introduced in Sections 19.1.2 and 19.1.3 as:

- **Near-term:** protoDUNE (the CERN-based prototype) and Wire Cell (event reconstruction software)
- **Long-term:** the full DUNE detector

Data-centric characteristics and requirements for both are presented in Table 19.2. It provides a brief high-level summary of the distinct stages of data lifetime: during data acquisition, near-term processing, and creation of metadata. There are important caveats in this table that need to be explained:

- The di erence in background conditions between protoDUNE and DUNE is tremendous (as already mentioned) due to the former being placed on Earth's surface and the latter at a significant depth inside a mine. For that reason, there is no real reason for the data rates and volumes to scale with the detector size or channel count, when looking at the data in the respective columns.
- The 53PB annual volume in DUNE is due to be revised downward by an order of magnitude, due to the following: most of these data would be due to the fraction of signals from ^{39}Ar decays which are above the ZS threshold chosen based on some metrics of signal-to-noise in the baseline design. It does not take into account the future capability of DAQ (now being developed) which will reject di use, very low energy signals lying outside of any regions of interest associated with localized activity.
- Not included in the experiment-side processing are express calibrations that will be necessary for monitoring and data QA purposes—this item is now in the initial stages of development and there is no metric yet available for reference.

While the simulation's dataflow was not included in this discussion for the sake of brevity, it is helpful to note that it will present an additional set of data-centric requirements such as considerable storage space and the efficient handling of metadata. Monte Carlo data in HEP experiments are typically larger in size than raw data, and this can be expected to be the case in DUNE due to necessary detailed studies of backgrounds and systematics etc. This will lead to creation of data sets of total size of perhaps O(PB).

In summary, both protoDUNE and DUNE will have challenging data-centric requirements at more than one level of its dataflow hierarchy. The most important are:

Processing stage	Present/Near-term (protoDUNE)	Long-term (DUNE)
Data acquisition rate: maximum rate(s) and totals	• 1 GB/s max. instantaneous • 200 MB/s sustained • ∼1 PB total for the run	• 1.7 GB/s sustained data rate • 53 PB annually
Experiment-side processing	• Online ZS, 10 GB/s ⇒ 1 GB/s • Compression	• Online ZS, 4.6 TB/s ⇒1.7 GB/s • Compression
Real-time constraints, turnaround time from collection to result for experimental control	• "Near-time" express analysis streams ∼O(10 min)	• "Near-time" express analysis streams, ∼O(10 min) • Supernova Burst trigger, decision time <1 s • 46 TB at full stream rate, recorded over ∼30 s.
Metadata/provenance capture	• Automatic tagging by online systems, reflecting run conditions.	• Automatic tagging by online systems, reflecting run conditions. • Tagging of readout frames of special interest based on DAQ indications.

Table 19.2: Summary of data-centric requirements.

protoDUNE

- Speedy analysis of incoming data for purposes of QA and the adjustment of reconstruction algorithms

DUNE

- Real-time data reduction beyond ZS by defeating di use persistent background from ^{39}Ar decays.
- Bu ering of data for SNB detection and the application of real-time algorithms to identify the signature of such an event.

19.2 Impediments, Gaps, Needs, Challenges

There are multiple issues that need to be resolved in the computing sector of DUNE that di er widely in nature, so the list below is meant to be an inclusive sample which contains items from di erent categories.

- Mechanisms of real-time data bu ering in DAQ systems for the purposes of recording SNB events (high bandwidth) are yet to be developed.
- Power and cooling requirements of DAQ systems located in the cavern of the DUNE LArTPC will be the limiting factor for real-time noise rejection, the quality of trigger decision and data reduc-

tion, which will have consequences for computing architectures downstream (offline). Low power consumption in front-end computing systems could mitigate this problem.

- Tomographic event reconstruction in LArTPC: the optimal choice of techniques and their application is a challenge.
- Potentially significant CPU and storage requirements for Monte Carlo studies to enable determination of the experiment's systematic errors, especially when utilizing sophisticated event reconstruction techniques.
- Implicit reliance on eventual HPC deployment for computationally intensive reconstruction techniques (e.g., Wire Cell). There is currently very little expertise in the Collaboration as a whole in the application of accelerators (GPU and others), message passing interface (MPI) and HPC in general for solving those problems where parallelization can be exploited. Application of advanced reconstruction techniques in real (or "near") time is impeded by difficulties in just-in-time scheduling of the available HPC resources.

Case Study 20

Open Numerical Laboratories

Alexander S. Szalay
The Johns Hopkins University

20.1 Science Use Case

20.1.1 Present or Near Term

There is an ongoing e ort to build an exascale computer—a substantial scale-up from current systems. Few codes scale to run well on the millions of cores available today. As fewer and fewer researchers will be able to use these ever larger systems efficiently, it will become increasingly important to create usable science products from numerical simulations accessible to a broader pool of users. Data products from the largest simulations must be released, shared, reanalyzed and archived over extended periods.

It will become increasingly important to create usable science products from numerical simulations accessible to a broader pool of users. Data products from the largest simulations must be released, shared, reanalyzed and archived over extended periods.

Indeed, scientists in many disciplines would like to compare the results of their experiments to data emerging from numerical simulations based on first principles. This requires not only that we can run sophisticated simulations and models, but that at least a selected subset of the results of these simulations are available publicly, through an easy-to-use portal. We have to turn the simulations into open numerical laboratories in which anyone can perform their own experiments. Integrating and comparing experiments to simulations is a non-trivial data management challenge. Not every data set from the simulations has the same lifecycle. Some results are just transient and need to be stored for a short period to analyze, while others will become community references, with a useful lifetime of a decade or more.

As we have learned over the years, once the data volume is too large, we have to move the analysis to the data rather than the traditional approach, which moved the data where our computers were. With these large data volumes one has to approach the data in a fully algorithmic fashion manual exploration of small (or large) files is no longer feasible, we need novel access methods to deal with scientific data at scale.

Some of our simulations have been run at various locations, at ORNL, on the Jaguar system, at the Texas Advanced Computing Center, and currently we are running experiments at NERSC. Furthermore, we have an ongoing collaboration and data exchanges with Los Alamos National Laboratory.

There are various data analysis tasks, which can be categorized by the access patterns to data. There are analyses requiring *global access*, where we have to touch every data item in a given snapshot. These typically require a facility that can keep the whole snapshot in memory, like a large 3D Fast Fourier Transform.

In order to perform the uncertainty quantification (UQ), we also need *ensemble access* to a potentially large number of simulations where the finite volume e ects can be averaged over, and we can see how results from simulations with identical physics, but di erent random numbers scatter.

There are analyses that are quite similar to the *rendering of a large subvolume*. This do not necessarily mean visualization, rather the collection of lower dimensional aggregates which are quite similar to projections onto a virtual "screen." Such patterns are typically well implemented in hardware accelerators, like GPUs.

Then we have *localized access*, where we need the data from the simulation in several small volumes, typically the size of an interpolation kernel. Here an efficient spatial indexing can have large implications for the speed of access.

Finally, a very scalable novel access method is where the users can insert *immersive virtual sensors* into the simulation. These sensors can then feed data back to the user. These sensors can provide a one-time measurement, they can be pinned to a physical (Eulerian) location or they can "go with the flow" as co-moving Lagrangian particles. In this case, assuming that the sensors can access the data server side quickly, the only scaling is related to the number of immersive particles.

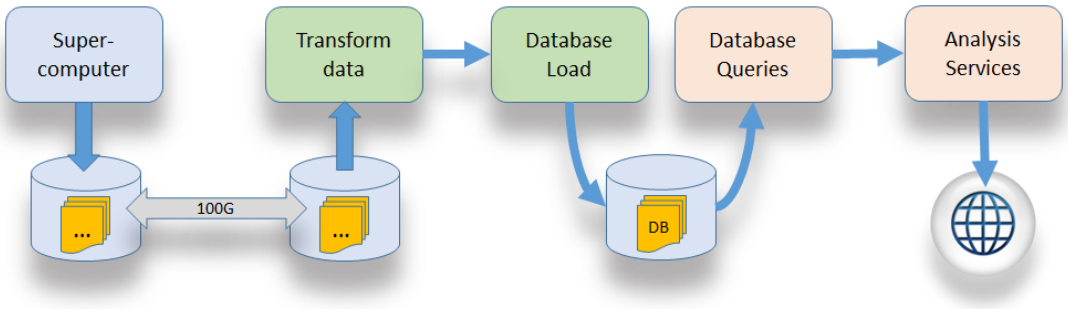


Figure 20.1: The simulation is run at a supercomputer, and the results are stored locally. The data is transferred to the site Open Numerical Laboratory site, and is transformed and indexed for database ingestion. Then the data and its indexes are loaded into a database. Finally the analysis services perform their data access through database queries, and perform additional value-added computations on the server side.

Today we use of the order of 64–128 snapshots of the simulations, typically containing on the order of a billion particles or grid points. This of course is not necessarily scalable as the memory footprint of future simulations will grow. For example, the Millennium XXL simulation with its 300 billion particles was only able to save 4 snapshots of the particle data, and had to settle on storing 64 snapshots of the much smaller subhalo catalogs. Our current databases range between 30 TB to 150 TB per simulation, although we have two simulations potentially exceeding a petabyte in the queue.

20.1.2 Future

Even though the largest simulations today are approaching hundreds of billions of particles or grid points, the total size of the output generated rarely exceeds 100 TB and almost never reaches a petabyte. There are many reasons for this. The larger the computer, the more cumbersome check—pointing becomes. Although the biggest machines have a terabyte-per-second of sequential bandwidth to the secondary storage, copying 100 TB takes too long to do frequently. In practice, this limits the number of snapshots captured. As the interconnect speeds are not going to increase by a factor of 30–100, it is likely that this limitation remains in place. Even with exascale machines, the outputs will remain in the range of a few petabytes.

When the primary consideration is restart, it is enough to have a small number of snapshots. On the other hand, if the goal is to be able to reconstruct the fine-grained spatial and temporal history of the simulation, and look at any part in detail, it is important to match the high-spatial resolution with an appropriate number of snapshots. For example, if we simulate a Milky-Way-like galaxy, and want to study its dynamics in detail in our laboratory, we need to save outputs more frequently than the rotational period of 108 years. This means that even the simulations need to be designed and run di erently if the target is to create a long-lived numerical laboratory used by hundreds of scientists.

As a result, in the exascale world only a small fraction of the complete output can ever be saved for later reuse and much of the analysis will have to be done *in situ*. If we cannot store all the data, it is of utmost importance to save at least the subsets with the highest information content and make these available for a wide audience.

. . . in the exascale world only a small fraction of the complete output can ever be saved for later reuse and much of the analysis will have to be done *in situ*.

Much of the scientific objective remains the same, except we have to make some very hard tradeo s in how we get there. Experimental particle physics has been forced to make these a decade ago. One can draw a good parallel to exascale simulations in order to predict what sort of thinking will be required.

At the Large Hadron Collider (LHC), the main facility is the collider ring, which provides several taps for the di erent experiments to place their detectors. At the detectors there is an enormous data rate. There are hardware triggers used to perform *in situ* computations on the data, and decide which events should be stored. These only fire for 1 out of 10,000,000 events, and the resulting data stream is still tens of petabytes.

The particle physicists would love to store all the event data, but they cannot. So they store the events with the greatest information content, a small enough fraction that their storage is still doable, but carefully selected so that all the important science can still be done without a compromise.

This is the way large simulations need to evolve towards: do as much as you can *in situ*, and then carefully select a small enough subset for *posterior* analyses that can be shared with a broader community.

We expect to work with several of the SC facilities, namely Los Alamos National Laboratory (LANL), ORNL, ANL and NERSC in the future. We currently have an active collaboration with Jim Ahrens at LANL.

Today we use typically about 100 snapshots out of thousands of timesteps taken. This is largely dictated from one end by the cost of checkpointing, from an other end about the details of the physical analyses, i.e. do we need to consider a temporal sequence of the snapshots, and interpolate across, or are they analyzed in isolation? In any case, we typically store all the data from a snapshot.

In the future, mostly we will not be able to do this. We will need to use clever machine learning algorithms to identify the scientifically most interesting (but small) localized regions, and save them at regular intervals, until they remain interesting. We will also need to save random samples of the volume, to be able to

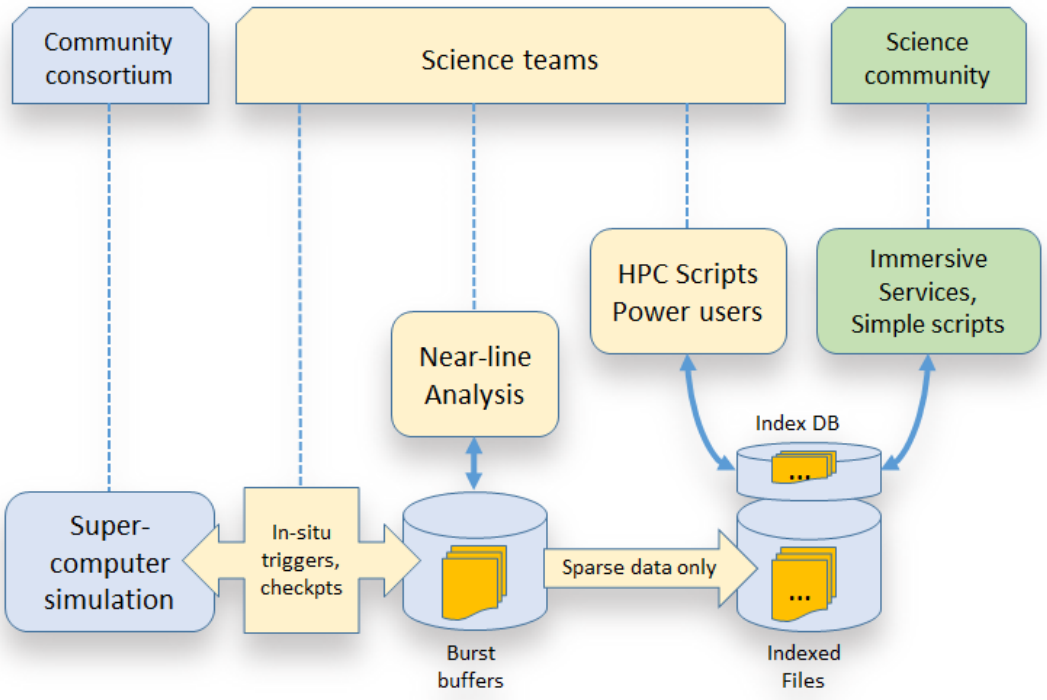


Figure 20.2: The simulation is defined by the whole community, and run at an exascale supercomputer. The participating science teams define a set of *in situ* triggers and tools. The output of the triggers and the checkpoints go to the local burst bu ers. The checkpoints and some of the triggered (sparse) outputs are analyzed there. The sparse (much smaller) data migrates to an analysis facility, consisting of an indexed file storage (FileDB) and an index database, for tracking the regions of interest. These data sets form the Open Numerical Laboratory, available by the whole community.

characterize the "boring," predictable parts of the simulation.

Finally in some cases we need to create a very frequent, but very localized data dump for some special science cases, like computing a "light-cone" in a cosmological simulation, where the di erent parts of the cone are seen by a distant observer at di erent times.

20.1.3 Data Lifecycle

- *In situ analysis*

  Most truly large numerical simulations are analyzed on the fly. The analysis tools are integrated with the simulation, and the derived data products are computed while the simulation is running. As these quantities represent only a small fraction of the data, it is easy to save these values often to disk. Full restart snapshots are thus only generated quite infrequently. The disadvantage is that if a new analysis idea emerges after the run is finished the whole simulation needs to be redone.

- *Private reuse*

  Sometimes a few tens of snapshots are saved to scratch disks, tightly coupled to the supercomputers,

and occasionally a segment of the simulation can be regenerated later from restarting from the nearest snapshot. This is typically done by the same team who ran the original simulation.

- *Public reuse*

  There are a few cases when the simulation outputs are made available. This is usually done through sharing the limited number of snapshot files. These are typically placed in a public file server, and can be downloaded at will. However, this practically limits data downloads to a few terabytes at most. The limiting factor is usually the network bandwidth, although the available storage at the user's end is also a problem.

- *Public service portal*

  In a few cases the simulation outputs are made available through publicly available services, enabling the users to perform either some extractions or computations over the data. This idea of "virtual data" has been around for more than a decade, but it has found limited uses. The Earth sciences community has used OpenDAP to expose large data sets (OpenDAP 2010) and enable a RESTful URL to subset and aggregate the data. In astrophysics, the Millennium Database has been the forerunner of such e orts. In this scenario, the creation of the public service portal and its complex functionalities requires a substantial e ort, thus it is only worth doing if the data set will remain public for an extended period—at least a few years.

> In a few cases the simulation outputs are made available through publicly available services, enabling the users to perform either some extractions or computations over the data. ...In astrophysics, the Millennium Database has been the forerunner of such e orts. In this scenario, the creation of the public service portal and its complex functionalities requires a substantial e ort, thus it is only worth doing if the data set will remain public for an extended period, at least a few years.

- *Archiving and long-term curation*

  There are very few data sets that have reached this stage of their lifecycle. Here, the biggest issue is that not every simulation will be equally used by the public, and over the years some of them will fade into irrelevancy while others emerge as a community reference. It is these latter simulations which need to be kept for a long time, even if just for comparison and reference. For such collections, used my many di erent refereed publications, reproducibility of these analyses will become another reason to keep the data, even when better and higher resolution alternatives become available. However, as the price of both storage and computation are expected to follow the current trend of becoming cheaper, these "legacy" data sets will comfortably fit in the shadow of the latest and best simulations.

> ...over the years some [data sets produced by simulation will] emerge as a community reference. ...For such collections, used by many di erent refereed publications, reproducibility of these analyses will become another reason to keep the data, even when better and higher resolution alternatives become available.

20.2 Impediments, Gaps, Needs, Challenges

There are several needs for this approach to be successful. These involve architectural components as well as software components.

Processing stage	Present/Near term	Long term
Data acquisition rate: maximum rate(s) and monthly or annual totals	10 GB/s maximum data rate; 500-800 TB annually	100 GB/s maximum data rate; 10-20 PB annually
Experiment-side processing	some *in-situ* analysis, some near-line posterior analysis	*in situ* triggers, *posterior* community analysis using immersive tools, comparisons to experimental data
Real-time constraints, turn around time from collection to result for experimental control	N/A today	15 mins for large-scale interactive analyses, 6 hours for batch jobs
Metadata/provenance capture	Limited info in file headers	Fully automated metadata and provenance and capture, stored in easy to search databases

Table 20.1: Summary of data-centric requirements for Open Numerical Laboratories.

- We need to define an easy-to-use generic API to plug in *in situ* triggering code into large simulations. These should enable the efficient use of machine learning, pattern recognition tools for identifying the regions of interest.
- Need for community-(or facility-)centric data repository for data archival, sharing; with substantial bandwidth to the stored data, and easy interface for interacting with the data analytics. This needs to be massively parallel, a combination of visualization and various analysis tools.
- Need for scalable analysis metaphors, like virtual sensors, immersive tools
- Currently we have insufficient ability to move, or very difficult to move data from experiment/acquisition to elsewhere for further processing, sharing, archival, etc.; we need a highly efficient multi-tier hierarchy for the analysis, consisting of burst bu ers, near-line storage, cold storage, with various levels of Amdahl numbers.

Case Study 21

DOE HEP Cosmic Frontier Use Cases

Salman Habib[3]

Contributors (from the ASCR/HEP Exascale Requirements Review):
S. Bailey[1], D. Bard[1], A. Borgland[2], J. Borrill[1], K. Heitmann[3], P. Nugent[1], N. Padmanabhan[4], D. Petravick[5]

[1] Lawrence Berkeley National Laboratory
[2] SLAC National Accelerator Laboratory
[3] Argonne National Laboratory
[4] Yale University
[5] National Center for Supercomputing Applications

The material presented here aims to concisely present data-related challenges in the DOE HEP Cosmic Frontier program. The information is largely abstracted from a number of recent planning meetings and requirements reviews. References are provided to enable access to broader background material as well as to more in-depth resources. Only the primary author is responsible for any accuracies or wrong opinions presented in the document.

21.1 Introduction

Scientific activities within DOE HEP can be broadly categorized within the Energy, Intensity, and Cosmic Frontiers. The three frontiers together address the following five science drivers recently identified in the P5 report [256]:

- Use the Higgs boson as a new tool for discovery
- Pursue the physics associated with neutrino mass
- Identify the new physics of dark matter
- Understand cosmic acceleration: dark energy and inflation
- Explore the unknown: new particles, interactions, and physical principles

The Energy and Intensity Frontiers are predominantly accelerator-based and focus on the physics of very small scales. The Cosmic Frontier covers two programs. The first focuses on the detection and mapping of galactic and extra-galactic sources of radiation utilizing a variety of well-instrumented telescopes, both ground- and satellite-based, with the purpose of a better understanding of the fundamental nature of the

dynamics and constituents of the Universe. The primary science thrusts within this frontier are understanding the nature of cosmic acceleration (investigating dark energy), discovering the origin and physics of dark matter, the dominant matter component in the universe, and investigating the nature of primordial fluctuations, which is also a test of the theory of inflation. In addition, these surveys can also provide unique probes of the sum of neutrino masses.

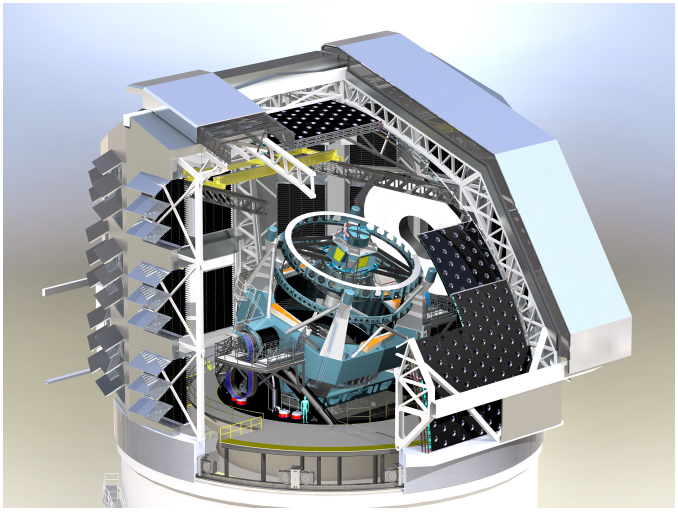


Figure 21.1: LSST (under construction): cutaway of the dome showing the telescope within.

A number of sky surveys in multiple wavebands are now scanning the sky to shed light on these problems, such as the Dark Energy Survey (DES) [257]. Near-future observations will be carried out by the Dark Energy Spectroscopic Instrument (DESI) [258] and the Large Synoptic Survey Telescope (LSST) [259] surveys in the optical, and by the CMB-S4 (Cosmic Microwave Background–Stage 4) survey [260] in the microwave band. These surveys will generate extremely large data sets in the hundreds of petabytes. Very large radio surveys, such as the Square Kilometer Array (SKA) [261], are also in planning stages, although there is no direct DOE (or indeed the United States) involvement at this point.

The second type of experiments under the Cosmic Frontier are dark matter detection experiments. These include direct detection experiments with cryogenic detectors (e.g., LZ [262] and SuperCDMS [263]) and indirect detection using high energy particles from space (e.g., Fermi [264] and HAWC [265]). Computational requirements in this sector are small to medium scale and do not reach the extreme requirements of large-scale sky surveys. Furthermore, the types of computational requirements are much closer to those for precision Intensity Frontier experiments (covered separately) than they are to sky surveys. Therefore the focus here will be on sky survey requirements.

21.2 Current Use Cases

21.2.1 Science Objectives and Motivations

The fundamental approach in sky surveys is to obtain wide and deep images of the sky, including spectra of selected objects, and to search for transient sources (e.g., supernovae). Depending on the nature of the survey (e.g., photometric vs. spectroscopic), the types of data and the required data pipelines, as well as the type of final analysis, can vary considerably. Despite the fact that a given survey may be focused on a few key science missions, usually a diverse set of science activities can be carried out with substantial discovery potential, since cosmological surveys are, by definition, surveys. This is to be contrasted with experiments

where a large amount of the data may have to be filtered or rejected in order to focus on a necessarily finite set of tasks.

Photometric surveys (e.g., DES) obtain galaxy images through a relatively modest number of filter bands (4–5). The main science goals of these surveys include measurements of gravitational weak and strong lensing, cluster abundance, supernova searches, and probes of galaxy clustering. Spectroscopic surveys obtain detailed spectra of a set of target galaxies (but they can also be done blind). This information enables the surveys to target 3D galaxy clustering probes and to study redshift space distortions. Because of the cost associated with redshift observations, photometric surveys are typically much deeper, and contain a much larger number of sources than spectroscopic surveys.

Surveys of the cosmic microwave background map the temperature and polarization of the microwave sky using ground-, balloon-, and space-borne instruments. With increases in sensitivity and resolution, these instruments can detect individual sources and objects (e.g., clusters). Furthermore, gravitational lensing of the CMB sky by the foreground matter distribution can be detected via polarization-sensitive instruments. This is useful in its own right as well as serving as a way to increase the sensitivity of searches for primordial gravitational waves.

21.2.2 User/Computing Facilities

The DOE HEP Cosmic Frontier program in cosmological surveys has involved partnerships with the National Aeronautics and Space Administration (NASA) and NSF. Unlike the Energy and Intensity Frontiers, DOE HEP does not run facilities analogous to LHC at CERN or the accelerator complex at Fermilab. Typically, the telescope, and the associated (on-site) first level of data acquisition and (o -site) data management pipelines, have been the responsibility of NSF (e.g., data handling for DES is the primary responsibility of the National Center for Supercomputing Applications, NCSA). DESI is a counter-example, with NERSC taking on this role. In future, given previous experience, it is likely that DOE facilities will take on a larger role in survey data management roles. Current DOE facilities involved in data analysis and management for surveys include BNL (DES), FNAL (DES), and LBNL/NERSC (CMB surveys, DES, DESI, LSST) and this number is likely to grow in the future to include the LCFs at Argonne and Oak Ridge.

21.2.3 Process of Science

The process of science involves, roughly speaking, a three-stage process, 1) data acquisition, 2) data processing, 3) data analysis. Typically, the science collaborations (which would be the analog of facility "users"), are handed "cleaned/calibrated" data as the end-point of the second stage. The first two stages are the responsibility of the project (analog of the "facility"), while the codes used for data analysis are the responsibility of the science collaborations or individual PIs once the data is made publicly available. In practice, the barrier between the the second and third stages is porous and considerable collaboration can take place between the production teams and the scientists. In any case, even the production pipelines (stage 2) are often built on community-provided tools. The LSST is an example of a survey that is attempting to do everything from scratch (unlike DES).

Stage 3 data analysis covers a multitude of tasks, such as galaxy shape measurements, the determination of photometric redshifts, transient searches, lensing shear determinations, and computation of various correlation functions and power spectra. For the CMB, timestream data is converted to maps, and the maps are then analyzed (which may involve cross-correlations with optical surveys) to extract science. Converting the stage 3 measurements into scientific inferences may involve another nontrivial computational step, based on either simulation or model-based approaches to statistical inverse problems (e.g., extraction of cosmological parameters).

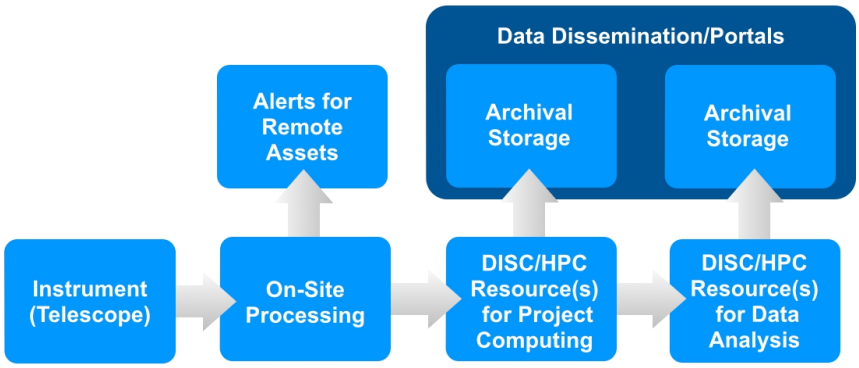


Figure 21.2: Notional data flow for a Cosmic Frontier survey experiment. On-site processing can be viewed as a medium-scale resource, augmented by specialized hardware/software for handling real-time tasks such as discovery alerts. Project and analysis computing can be handled at multiple sites; DISC (Data-Intensive Scalable Computing) resources include clusters, clouds, and HTC (High Throughput Computing) systems. Data can be archived and served from multiple sites. Both single and multi-user modes need to be supported.

Stages 1 and 2 of this process may be thought of jointly as data acquisition, data management, and data curation. Although these steps are important, they do not consume the majority of computational resources (for DES, this is roughly the equivalent of about 1 million core-hours annually). The analysis component is the main consumer of resources (by more than an order of magnitude), and is itself dwarfed by the simulation requirements (another order of magnitude).

Note that unlike some other use cases, there is very little of a feedback loop in survey observations—if something interesting is found, it is usually the task of other assets to carry out follow-up observations.

21.2.4 Extent of Data Use

Survey data is very valuable with a very long shelf-life and is mined and analyzed in a number of ways, depending on the science use case. Currently, very little data is left on the floor, as the data acquisition rates are not a problem (the exception here are large radio surveys, where only a small subset comprising of processed data can be stored on disk). Most surveys archive all the data taken, at least all that meets certain quality cuts, and make the data public. One case where data loss occurs is in studies of transients because of possible inefficiencies in the detection technology, classification algorithms, and lack of follow-up resources. Other issues that prevent making use of the complete data set are technical issues such as lack of understanding of foregrounds, modeling the atmosphere, detector noise, etc.

The total raw data sizes range from small/medium scale (about 100TB–1PB) to large (about 100PB). However, a large fraction of the science analyses work with reduced data sets—catalogs of objects. These reduced catalogs, even in the most extreme cases, are unlikely to be larger than about 1 PB in size, i.e., roughly two orders of magnitude smaller than the largest of the raw data sets. At the same time, it is important to keep in mind that new science cases often emerge as computational capabilities improve, so one should view these numbers more as a statement of boundary conditions set by resource restrictions, rather than an absolute estimate. It is quite possible for the derived data sets to be significantly larger in the future.

21.3 Future Use Cases

21.3.1 Science Objectives and Motivations

Surveys operate on relatively long timescales, from a few years to decades (for example, the Sloan Digital Sky Survey, first light in 1998 [266], is still operating, albeit with changing scientific *focii*). The general scientific directions and observational technologies for the Cosmic Frontier are more or less well defined on the timescale of the next decade, and have been listed in the previous section. Beyond that a number of possible new technologies may enter survey planning (e.g., 21cm, fast spectroscopic methods) but it is probably too early to speculate on the needs this far out. Future directions in the field are set by community consensus and funding agency priorities, typically on a decadal timescale.

21.3.2 User and Computing Facilities

It is unlikely that the current model will change very much on the timescale of the next decade. As stated earlier, it is very likely that DOE facilities (both ASCR and HEP) will take on a significantly larger role in data archiving, transfer, and analysis. It is also possible that commercial cloud resources will become a major resource in these areas—although several outstanding questions remain (e.g., cost models, data archiving and transfer); this disruptive possibility needs to be continuously explored. The main new hardware trend of interest for DOE facilities—in the relatively near-term—is the evolution and integration of HPC systems within a data-centric usage model.

> The main new hardware trend of interest for DOE [science user] facilities—in the relatively near-term—s the evolution and integration of HPC systems within a data-centric usage model.

21.3.3 Process of Science

The actual process of science is unlikely to change in any significant manner over and above the current paradigm (large projects or collaborations) and increased public data access. One of the possible changes in the mode of operation is the continuing movement of available computational resources from local to remote facility-based (moving computing to the data). It is doubtful, however, that this change in the underlying support technology will actually lead to a major change in the underlying scientific process.

21.3.4 Extent of Data Use

The extent of data use is likely to evolve somewhat as better and larger computational resources become available, however, it is unlikely to change radically. The basic throughput requirement is essentially set by the data rate at the detectors, since the sky is not a high-intensity source, data rates are automatically limited. Even with LSST, the data rate is only about 15TB/night (CMB-S4 would be roughly 1TB/day), which in the early 2020s, is a very modest target requirement. Thus, the hope is that a very large fraction of the data generated by the instruments would be usable data.

21.3.5 Data Lifecycle

Details of the data lifecycle vary considerably in implementation but more or less follow the description given in Section 21.2.3 (Cf. Figure 21.2). Individual experiments have widely di erent on-site computing requirements (from the "power workstation" to cluster level), but it is generally true that o -site computing and data-related requirements are significantly larger in all cases.[1]

21.4 Impediments, Gaps, Needs, Challenges

There are a number of areas for future work that have been identified in several studies carried out by the HEP community. Here we list a set of the identified issues; more details can be found in a number of references—the Snowmass Summer Studies [271], ASCR/HEP Data Summit Report [272], ASCR/HEP Exascale Requirements Review report [273], and the HEP-FCE Working Groups report [274].

Some of the issues that have been raised so far are:

- Lack of trained manpower and career paths for computationally-oriented scientists;
- Integration of automated data transfer, storage, and archiving within scientific workflows;
- Lack of standard data formats and data structures—data organization and data structures to optimize data selection, manipulation, and analysis;
- Scalable and approximate algorithms for data analysis (e.g., anomaly detection, clustering); associated uncertainty quantification (UQ) and verification and validation (V&V);
- Increased interactive access with data, including sophisticated analyses, use of cloud-like services (e.g., portals);
- Evolution of software stack for next-generation architectures;
- Machine learning/statistics methods for classification, regression, and solution of high-dimensional inverse problems; associated UQ and V&V; and
- Data archiving and curation strategies.

It is important to keep in mind that programmatic HEP activities are almost never "single investigator." The projects are usually broad and involve a number of potentially complex issues. The HEP Energy and Intensity Frontier communities cover essentially everyone in the entire field. While this is not true in the Cosmic Frontier, which is a subset of a broader astronomy/physics e ort, it is still the case that this is a large subset. Consequently, solutions to problems faced by this community will be quickly adopted by a much broader set of scientific activities.

[1] An overview descriptions of individual implementations and data flow organization are given in the following references: DES [267], LSST [268], DESI [269], and CMB [270].

Bibliography

[1] B. Obama, "Executive Order – Creating a National Strategic Computing Initiative," Jul. 2015, https://www.whitehouse.gov/the-press-office/2015/07/29/executive-order-creating-national-strategic-computing-initiative, last accessed March 2016.

[2] "FACT SHEET: National Strategic Computing Initiative," Jul. 2015, https://www.whitehouse.gov/sites/default/files/microsites/ostp/nsci_fact_sheet.pdf, last accessed March 2016.

[3] R. Weiss and L.-J. Zgorski, "Obama Administration Unveils "Big Data" Initiative: Announces $200 Million in New R&D Investments," Mar. 2012, https://www.whitehouse.gov/sites/default/files/microsites/ostp/big_data_press_release_final_2.pdf, last accessed March 2016.

[4] T. Kalil and F. Zhao, "Unleasing the Power of Big Data," Apr. 2013, https://www.whitehouse.gov/blog/2013/04/18/unleashing-power-big-data, last accessed March 2016.

[5] J. Cummings, T. Finholt, I. Foster, C. Kesselman, and K. A. Lawrence, "Beyond being there: A blueprint for advancing the design, development, and evaluation of virtual organizations," 2008, http://web.ci.uchicago.edu/events/VirtOrg2008/VO_report.pdf.

[6] T. Hey, S. Tansley, and K. Tolle, *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft Research, 2009.

[7] "Scientific collaborations for extreme-scale science," 2011, http://www.orau.gov/ASKD2013/2011rpt.pdf.

[8] "Accelerating Scientific Knowledge Discovery (ASKD) Working Group Report," 2013, http://www.orau.gov/askd2013/ASKD_Report_V1_0.pdf.

[9] D. Agarwal, A. Boehnlein, R. Carlson, M. Ernst, I. Foster, B. Jennings, S. Klasky, K. Kleese Van Dam, R. Pordes, and D. Skinner, "Vision for an Accelerating Scientific Knowledge Discovery (ASKD) Program," 2013, http://www.orau.gov/ASKD2013/vision.pdf.

[10] J. Gray, A. Szalay, A. Thakar, P. Z. Kunszt, T. Malik, J. Raddick, C. Stoughton, and J. vandenBerg, "The SDSS SkyServer – public access to the Sloan Digital Sky Server data," in *ACM SIGMOD*, Year, pp. 1–11.

[11] R. A. Overbeek, T. Disz, and R. L. Stevens:, "The SEED: A peer-to-peer environment for genome annotation," *Communications of the ACM*, vol. 47, no. 11, pp. 46–51, 2004.

[12] R. Overbeek, R. Olson, G. D. Pusch, G. J. Olsen, J. J. Davis, T. Disz, R. A. Edwards, S. Gerdes, B. Parrello, M. Shukla, V. Vonstein, A. R. Wattam, F. Xia, and R. Stevens, "The seed and the rapid annotation of microbial genomes using subsystems technology (rast)," *Nucleic Acids Research*, vol. 42, no. D1, pp. D206–D214, 2014.

[13] F. Meyer, D. Paarmann, M. D'Souza, R. Olson, E. Glass, M. Kubal, T. Paczian, A. Rodriguez, R. Stevens, A. Wilke, J. Wilkening, and R. Edwards, "The metagenomics RAST server – a public resource for the

automatic phylogenetic and functional analysis of metagenomes," *BMC Bioinformatics*, vol. 9, no. 1, p. 386, 2008.

[14] W. O'Mullane, N. Li, M. Nieto-Santisteban, A. Szalay, A. Thakar, and J. Gray, "Batch is back: Casjobs, serving multi-tb data on the web," in *Web Services, 2005. ICWS 2005. Proceedings. 2005 IEEE International Conference on*. IEEE, 2005, pp. 33–40.

[15] I. Foster, "Globus Online: Accelerating and democratizing science through cloud-based services," *IEEE Internet Computing*, no. May/June, pp. 70–73, 2011.

[16] K. Chard, S. Tuecke, and I. Foster, "Efficient and secure transfer, synchronization, and sharing of big data," *Cloud Computing, IEEE*, vol. 1, no. 3, pp. 46–55, 2014.

[17] D. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 6–23, 1997.

[18] D. Lahat, T. Adali, and C. Jutten, "Multimodal data fusion: An overview of methods, challenges, and prospects," *Proceedings of the IEEE*, vol. 103, no. 9, pp. 1449–1477, 2015.

[19] Y. Xiong, D. Wang, Y. Zhang, S. Feng, and G. Wang, "Multimodal data fusion in text-image heterogeneous graph for social media recommendation," *LNCS Web-Age Information Management: 15th International Conference 2014*, vol. 8485, pp. 96–99, 2014.

[20] Committee on Mathematical Foundations of Verification, Validation, and Uncertainty Quantification; Board on Mathematical Sciences and Their Applications, Division on Engineering and Physical Sciences, National Research Council, *Assessing the Reliability of Complex Models: Mathematical and Statistical Foundations of Verification, Validation, and Uncertainty Quantification*. The National Academies Press, 2012. [Online]. Available: http://www.nap.edu/openbook.php?record_id=13395

[21] R. C. Smith, *Uncertainty Quantification: Theory, Implementation, and Applications*. SIAM, 2014.

[22] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed. New York: Springer Science & Business Media, 2009.

[23] H. Owhadi, C. Scovel, and T. Sullivan, "On the brittleness of Bayesian inference," *SIAM Review*, vol. 57, no. 4, pp. 566–582, 2015.

[24] H. Owhadi, C. Scovel, T. J. Sullivan, M. McKerns, and M. Ortiz, "Optimal uncertainty quantification," *SIAM Review*, vol. 55, no. 2, pp. 271–345, 2013.

[25] A. C. Bovik, *Handbook of Image and Video Processing (Communications, Networking and Multimedia)*. Orlando, FL: Academic Press, 2005.

[26] L. Ning, T. T. Georgiou, A. Tannenbaum, and S. P. Boyd, "Linear models based on noisy data and the Frisch scheme," *SIAM Review*, vol. 57, no. 2, pp. 167–197, 2015.

[27] M. Lebrun, M. Colom, A. Buades, and J. M. Morel, "Secrets of image denoising cuisine," *Acta Numerica*, vol. 21, pp. 475–576, 5 2012.

[28] S. van Buuren, *Flexible Imputation of Missing Data*. Boca Raton, FL: Chapman and Hall/CRC, 2012.

[29] P. Benner, V. Mehrmann, and D. C. Sorensen, *Dimension Reduction of Large-Scale Systems*. Springer-Verlag Berlin Heidelberg, 2005, vol. 35.

[30] B. Clarke, E. Fokoué, and H. H. Zhang, *Principles and Theory for Data Mining and Machine Learning*. Springer-Verlag New York, 2009.

[31] Y.-X. Wang and Y.-J. Zhang, "Nonnegative matrix factorization: A comprehensive review," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 6, pp. 1336–1353, 2013.

[32] R. Kannan, G. Ballard, and H. Park, "A high-performance parallel algorithm for nonnegative matrix factorization," in *Principles and Practice of Parallel Programming (PPoPP)*, 2016.

[33] M. W. Mahoney and P. Drineas, "CUR matrix decompositions for improved data analysis," *Proceedings of the National Academy of Sciences*, vol. 106, no. 3, pp. 697–702, 2009.

[34] K. Huang, N. D. Sidiropoulos, and A. P. Liavas, "A flexible and efficient algorithmic framework for constrained matrix and tensor factorization," *arXiv preprint arXiv:1506.04209*, 2015.

[35] W. Hackbusch, "Numerical tensor calculus," *Acta Numerica*, vol. 23, pp. 651–742, 5 2014.

[36] W. Austin, G. Ballard, and T. G. Kolda, "Parallel tensor compression for large-scale scientific data," in *IPDPS'16: Proceedings of the 30th IEEE International Parallel & Distributed Processing Symposium*, May 2016.

[37] D. F. Gleich, "PageRank beyond the web," *SIAM Review*, vol. 57, no. 3, pp. 321–363, 2015.

[38] D. F. Gleich, L.-H. Lim, and Y. Yu, "Multilinear PageRank," *SIAM Journal on Matrix Analysis and Applications*, vol. 36, no. 4, pp. 1507–1541, 2015.

[39] G. Carlsson, "Topology and data," *Bulletin of the American Mathematical Society*, vol. 46, no. 2, pp. 255–308, 2009.

[40] C. C. Aggarwal, *Data Streams: Models and Algorithms*. Springer Science & Business Media, 2007, vol. 31.

[41] A. Broder and M. Mitzenmacher, "Network applications of bloom filters: A survey," *Internet Mathematics*, vol. 1, no. 4, pp. 485–509, 2004.

[42] G. Carlsson, "Topological pattern recognition for point cloud data," *Acta Numerica*, vol. 23, pp. 289–368, 5 2014.

[43] T. dos Santos Rolo, A. Ershov, T. van de Kamp, and T. Baumbach, "In vivo x-ray cine-tomography for tracking morphological dynamics," *Proceedings of the National Academy of Sciences*, no. 11, Mar. 2014.

[44] E. J. Candés, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[45] K. Bryan and T. Leise, "Making do with less: An introduction to compressed sensing," *SIAM Review*, vol. 55, no. 3, pp. 547–566, 2013.

[46] A. Tripathi, I. McNulty, and O. G. Shpyrko, "Ptychographic overlap constraint errors and the limits of their numerical recovery using conjugate gradient descent methods," *Optics Express*, vol. 22, no. 2, pp. 1452–1466, 2014.

[47] M. N. Garofalakis and P. B. Gibbons, "Approximate query processing: Taming the TeraBytes," in *VLDB*, 2001.

[48] J. Wang, S. Krishnan, M. J. Franklin, K. Goldberg, T. Kraska, and T. Milo, "A sample-and-clean framework for fast and accurate query processing on dirty data," in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*. ACM, 2014, pp. 469–480.

[49] [Online]. Available: http://numba.pydata.org

[50] [Online]. Available: https://www.txcorp.com/gpulib

[51] [Online]. Available: https://m.vtk.org

[52] M. Bauer, S. Treichler, E. Slaughter, and A. Aiken, "Legion: Expressing locality and independence with logical regions," in *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*. IEEE Computer Society Press, 2012, p. 66.

[53] H. C. Edwards and C. R. Trott, "Kokkos: Enabling performance portability across manycore architectures," *In Extreme Scaling Workshop (XSW)*, Aug. 2013.

[54] C. Sewell, J. Meredith, K. Moreland, T. Peterka, D. DeMarle, L.-t. Lo, J. Ahrens, R. Maynard, and B. Geveci, "The sdav software frameworks for visualization and analysis on next-generation multicore and many-core architectures," in *Proceedings of the 2012 SC Companion: High Performance Computing, Networking Storage and Analysis*, ser. SCC '12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 206–214.

[55] D. S. Katz, G. Allen, N. C. Hong, K. Cranston, M. Parashar, D. Proctor, M. Turk, C. C. Venters, and N. Wilkins-Diehr, "Second workshop on sustainable software for science: Practice and experiences (wssspe2): Submission, peer-review and sorting process, and results," *arXiv preprint arXiv:1411.3464*, 2014.

[56] R. Brun and F. Rademakers, "Rootan object oriented data analysis framework," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 389, no. 1, pp. 81–86, 1997.

[57] L. K. Berg, L. D. Riihimaki, Y. Qian, H. Yan, and M. Huang, "The low-level jet over the southern great plains determined from observations and reanalyses and its impact on moisture transport," *Journal of Climate*, vol. 28, no. 17, pp. 6682–6706, 2015.

[58] K. Heitmann, N. Frontiere, C. Sewell, S. Habib, A. Pope, H. Finkel, S. Rizzi, J. Insley, and S. Bhattacharya, "The q continuum simulation: Harnessing the power of gpu accelerated supercomputers," *arXiv preprint arXiv:1411.3396*, 2014.

[59] Y. S. Nashed, D. J. Vine, T. Peterka, J. Deng, R. Ross, and C. Jacobsen, "Parallel Ptychographic Reconstruction," *Optics Express*, vol. 22, no. 26, pp. 32 082–32 097, 2014.

[60] J. Deng, Y. S. Nashed, S. Chen, N. W. Phillips, T. Peterka, R. Ross, S. Vogt, C. Jacobsen, and D. J. Vine, "Continuous motion scan ptychography: characterization for increased speed in coherent x-ray imaging," *Optics express*, vol. 23, no. 5, pp. 5438–5451, 2015.

[61] S. Chen, J. Deng, D. Vine, Y. Nashed, Q. Jin, T. Peterka, C. Jacobsen, and S. Vogt, "Simultaneous x-ray nano-ptychographic and fluorescence microscopy at the bionanoprobe," in *SPIE Optical Engineering+ Applications*. International Society for Optics and Photonics, 2015, pp. 95 920I–95 920I.

[62] R. Archibald, K. Evans, and A. Salinger, "Accelerating time integration for the shallow water equations on the sphere using gpus," *Procedia Computer Science*, vol. 51, pp. 2046–2055, 2015.

[63] J. Dean and S. Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters," *Commun. ACM*, vol. 51, pp. 107–113, January 2008. [Online]. Available: http://doi.acm.org/10.1145/1327452.1327492

[64] T. Peterka and R. Ross, "Versatile Communication Algorithms for Data Analysis," in *EuroMPI Special Session on Improving MPI User and Developer Interaction IMUDI'12*, Vienna, AT, 2012.

[65] D. Morozov and T. Peterka, "DIY2: Data-Parallel Out-of-Core Library," in *Submitted to ACM Symposium on Principles and Practice of Parallel Programming (PPoPP) 2016*, Barcelona, Spain, 2016.

[66] various. Catalog of data analysis software. [Online]. Available: https://www1.aps.anl.gov/Science/Scientific-Software

[67] J. Heinrich and D. Weiskopf, "State of the art of parallel coordinates," in *Eurographics 2013 - State of the Art Reports*, M. Sbert and L. Szirmay-Kalos, Eds. The Eurographics Association, 2013.

[68] D. Rogers and C. Garasi, "Prism: A multi-view visualization tool for multi-physics simulation," *Coordinated and Multiple Views in Exploratory Visualization, International Conference on*, vol. 0, pp. 85–95, 2005.

[69] R. Mortier, H. Haddadi, T. Henderson, D. McAuley, and J. Crowcroft, "Human-data interaction: The human face of the data-driven society," 2014.

[70] "The luajit project," http://luajit.org/, accessed: 2016-03-17.

[71] Z. DeVito, J. Hegarty, A. Aiken, P. Hanrahan, and J. Vitek, "Terra: a multi-stage language for high-performance computing," in *ACM SIGPLAN Notices*, vol. 48, no. 6. ACM, 2013, pp. 105–116.

[72] P. S. McCormick, J. Inman, J. P. Ahrens, C. Hansen, and G. Roth, "Scout: A hardware-accelerated system for quantitatively driven visualization and analysis," in *Visualization, 2004. IEEE*. IEEE, 2004, pp. 171–178.

[73] Z. DeVito, N. Joubert, F. Palacios, S. Oakley, M. Medina, M. Barrientos, E. Elsen, F. Ham, A. Aiken, K. Duraisamy *et al.*, "Liszt: a domain specific language for building portable mesh-based pde solvers," in *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*. ACM, 2011, p. 9.

[74] P. Kogge, K. Bergman, S. Borkar, D. Campbell, W. Carlson, W. Dally, M. Denneau, P. Franzon, W. Harrod, K. Hill, J. Hiller, S. Karp, S. Keckler, D. Klein, R. Lucas, M. Richards, A. Scarpelli, S. Scott, A. Snavely, T. Sterling, R. S. Williams, and K. Yelick, "Exascale computing study: Technology challenges in achieving exascale systems," DARPA Information Processing Techniques Office, Tech. Rep., 2008.

[75] A. Ananthaswamy, "Rise of the robot astronomers," *New Scientist*, vol. 209, no. 2795, pp. 20–21, 2011.

[76] A. Gal-Yam, M. M. Kasliwal, I. Arcavi, Y. Green, O. Yaron, S. Ben-Ami, D. Xu, A. Sternberg, R. M. Quimby, S. R. Kulkarni *et al.*, "Real-time detection and rapid multiwavelength follow-up observations of a highly subluminous type ii-p supernova from the palomar transient factory survey," *The Astrophysical Journal*, vol. 736, no. 2, p. 159, 2011. [Online]. Available: http://stacks.iop.org/0004-637X/736/i=2/a=159

[77] J. S. Vetter, Ed., *Contemporary High Performance Computing: From Petascale Toward Exascale*, 1st ed. Boca Raton: Taylor and Francis, 2015, vol. 2.

[78] J. S. Vetter and S. Mittal, "Opportunities for nonvolatile memory systems in extreme-scale high performance computing," *Computing in Science and Engineering special issue*, vol. 17, no. 2, pp. 73–82, 2015.

[79] J. Ang, K. Bergman, S. Borkar, W. Carlson, L. Carrington, G. Chiu, R. Colwell, W. Dally, J. Dongarra, A. Geist, G. Grider, R. Haring, J. Hittinger, A. Hoisie, D. Klein, P. Kogge, R. Lethin, R. Lucas, V. Sarkar, R. Schreiber, J. Shalf, T. Sterling, and R. Stevens, "Top ten exascale research challenges," DOE Office of Science, Advanced Scientific Computing Advisory Committee, Subcommittee for the Top Ten Exascale Research Challenges, Tech. Rep., 2014.

[80] Robinhood policy engine. [Online]. Available: https://github.com/cea-hpc/robinhood/wiki

[81] [Online]. Available: https://irods.org

[82] [Online]. Available: https://sdm.lbl.gov/srm-wg/doc/SRM.v2.2.html

[83] F. Donno, L. Abadie, P. Badino, J. P. Baud, E. Corso, S. De Witt, P. Fuhrmann, J. Gu, B. Koblitz, S. Lemaitre, M. Litmaath, D. Litvintsev, G. Lo Presti, L. Magnoni, G. McCance, T. Mkrtchan, R. Mollon, V. Natarajan, T. Perelmutov, D. Petravick, A. Shoshani, A. Sim, D. Smith, P. Tedesco, and R. Zappi, "Storage Resource Manager version 2.2: design, implementation, and testing

experience," CERN, Geneva, Tech. Rep. CERN-IT-Note-2007-031, Oct 2007. [Online]. Available: https://cds.cern.ch/record/1065764

[84] A. Klimentov and et al., "Atlas data processing and supercomputers, integration of panda workload management system with supercomputers," *XXV International Symposium on Nuclear Electronics and Computing*, 2015.

[85] Lingerfelt and et al., "Beam: Bellerophon environment for analysis of materials," *Joint NSRC Workshop 2015: Big, Deep, and Smart Data Analytics in Materials Imaging*, 2015. [Online]. Available: https://www.youtube.com/watch?v=LrPqZx2jazM

[86] A. Collaboration, "Data federation strategies for atlas using xrootd," *Journal of Physics: Conference Series*, vol. 513, no. 4, p. 042049, 2014. [Online]. Available: http://stacks.iop.org/1742-6596/513/i=4/a=042049

[87] K. B. for the CMS Collaboration, "Cms use of a data federation," *Journal of Physics: Conference Series*, vol. 513, no. 4, p. 042005, 2014. [Online]. Available: http://stacks.iop.org/1742-6596/513/i=4/a=042005

[88] Docker (software). [Online]. Available: https://en.wikipedia.org/wiki/Docker_(software)

[89] Investigating hpc infrastructure and user service provisioning through linux containers (lxc). [Online]. Available: http://jlse.anl.gov/projects/

[90] Cosmology incite project. [Online]. Available: http://cades.ornl.gov/science.html

[91] D. M. Jacobsen and R. S. Canon, "Contain this, unleashing docker for hpc," *Proceedings of CUG 2015*, 2015. [Online]. Available: https://cug.org/proceedings/cug2015_proceedings/includes/files/pap157.pdf

[92] [Online]. Available: http://fasterdata.es.net/science-dmz

[93] [Online]. Available: http://monalisa.cern.ch/FDT/sc09.html

[94] [Online]. Available: http://supercomputing.caltech.edu/tools.html

[95] "The future of scientific workflows: Report of the doe ngns/cs scientific workflows workshop april 20-21, 2015," ***FIX***, 2015.

[96] W. M. van Der Aalst, A. H. Ter Hofstede, B. Kiepuszewski, and A. P. Barros, "Workflow patterns," *Distributed and parallel databases*, vol. 14, no. 1, pp. 5–51, 2003.

[97] L. Ramakrishnan and B. Plale, "A multi-dimensional classification model for scientific workflow characteristics," in *Proceedings of the 1st International Workshop on Workflow Approaches to New Data-centric Science*. ACM, 2010, p. 4.

[98] I. Altintas, C. Berkley, E. Jaeger, M. Jones, B. Ludscher, and S. Mock, "Kepler: An Extensible System for Design and Execution of Scientific Workflows," 2004, citeseer.ist.psu.edu/altintas04kepler.html.

[99] B. Ludscher, I. Altintas, C. Berkley, D. Higgins, E. Jaeger, M. Jones, E. Lee, J. Tao, and Y. Zhao, "Scientific Workflow Management and the Kepler System," 2005, citeseer.ist.psu.edu/ludscher05scientific.html.

[100] T. Oinn, M. Greenwood, M. Addis, M. N. Alpdemir, J. Ferris, K. Glover, C. Goble, A. Goderis, D. Hull, D. Marvin, P. Li, P. Lord, M. R. Pocock, M. Senger, R. Stevens, A. Wipat, and C. Wroe, "Taverna: Lessons in Creating a Workflow Environment for the Life Sciences: Research Articles," *Concurr. Comput. : Pract. Exper.*, vol. 18, no. 10, pp. 1067–1100, 2006.

[101] E. Deelman, J. Blythe, Y. Gil, and C. Kesselman, "Workflow Management in GriPhyN," Grid Resource Management, J. Nabrzyski, J. Schopf, and J. Weglarz editors, Kluwer, 2003.

[102] D. Churches, G. Gombas, A. Harrison, J. Maassen, C. Robinson, M. Shields, I. Taylor, and I. Wang, "Programming Scientific and Distributed Workflow with Triana Services," *Concurrency and Computation: Practice and Experience (Special Issue: Workflow in Grid Systems)*, vol. 18, no. 10, pp. 1021–1037, 2006.

[103] L. Ramakrishnan, S. Poon, V. Hendrix, D. Gunter, G. Pastorello, and D. Agarwal, "Experiences with user-centered design for the tigres workflow api," in *e-Science (e-Science), 2014 IEEE 10th International Conference on*, vol. 1, Oct 2014, pp. 290–297.

[104] V. Hendrix, J. Fox, and L. Ramakrishnan, "Tigres workflow library: Supporting scientific pipelines on hpc systems," in *16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, May 2016.

[105] R. Yu, Jiaand Buyya, "A Taxonomy of Scientific Workflow Systems for Grid Computing," *SIGMOD Rec.*, vol. 34, no. 3, pp. 44–49, September 2005.

[106] D. Thain, J. Bent, A. C. Arpaci-Dusseau, R. H. Arpaci-Dusseau, and M. Livny, "Pipeline and batch sharing in grid workloads," in *High Performance Distributed Computing, 2003. Proceedings. 12th IEEE International Symposium on*. IEEE, 2003, pp. 152–161.

[107] I. J. Taylor, E. Deelman, D. B. Gannon, and M. Shields, *Workflows for e-Science: Scientific Workflows for Grids*. Springer, December 2006.

[108] Y. Simmhan, P. Groth, and L. Moreau, "Special section: The third provenance challenge on using the open provenance model for interoperability," *Future Generation Computer Systems*, vol. 27, no. 6, pp. 737 – 742, 2011.

[109] B. Cao, B. Plale, G. Subramanian, E. Robertson, and Y. Simmhan, "Provenance information model of karma version 3," in *Services - I, 2009 World Conference on*, july 2009, pp. 348 –351.

[110] Y. Simmhan, B. Plale, and D. Gannon, "A framework for collecting provenance in data-centric scientific workflows," in *Web Services, 2006. ICWS'06. International Conference on*. IEEE, 2006, pp. 427–436.

[111] L. M. G. Jr., B. Cli ord, M. Mattoso, M. Wilde, and I. Foster, "Provenance management in swift," *Future Generation Computer Systems*, vol. 27, no. 6, pp. 775 – 780, 2011.

[112] S. B. Davidson and J. Freire, "Provenance and scientific workflows: challenges and opportunities," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, ser. SIGMOD '08. New York, NY, USA: ACM, 2008, pp. 1345–1350, http://doi.acm.org/10.1145/1376616.1376772.

[113] M. Anand, S. Bowers, I. Altintas, and B. Ludscher, "Approaches for exploring and querying scientific workflow provenance graphs," in *Provenance and Annotation of Data and Processes*, ser. Lecture Notes in Computer Science, D. McGuinness, J. Michaelis, and L. Moreau, Eds. Springer Berlin / Heidelberg, vol. 6378, pp. 17–26, http://dx.doi.org/10.1007/978-3-642-17819-1_3.

[114] Y. L. Simmhan, S. L. Pallickara, N. N. Vijayakumar, and B. Plale, "Data management in dynamic environment-driven computational science," in *Grid-based problem solving environments*. Springer, 2007, pp. 317–333.

[115] C. R. Aragon, S. J. Bailey, S. Poon, K. Runge, and R. C. Thomas, "Sunfall: a collaborative visual analytics system for astrophysics," in *Journal of Physics: Conference Series*, vol. 125, no. 1. IOP Publishing, 2008, p. 012091.

[116] "myExperiment Workflow Archive," http://www.myexperiment.org/workflows.

[117] "Pegasus Workflow Archive," www.workflowarchive.org.

[118] "Report of the department of energy workshop on: Data and communications in basic energy sciences: Creating a pathway for scientific discovery," http://science.energy.gov/~/media/ascr/pdf/research/scidac/ASCR_BES_Data_Report.pdf, 2012.

[119] P. Nugent, Y. Cao, and M. Kasliwal, "The palomar transient factory," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2015, pp. 939 702–939 702.

[120] C. Docan, M. Parashar, and S. Klasky, "Dataspaces: an interaction and coordination framework for coupled simulation workflows," *Cluster Computing*, vol. 15, no. 2, pp. 163–181, 2012.

[121] H. Sim, Y. Kim, S. S. Vazhkudai, D. Tiwari, A. Anwar, A. R. Butt, and L. Ramakrishnan, "Analyzethis: an analysis workflow-aware storage system," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. ACM, 2015, p. 20.

[122] K. Wu, S. Ahern, E. W. Bethel, J. Chen, H. Childs, E. Cormier-Michel, C. Geddes, J. Gu, H. Hagen, B. Hamann *et al.*, "FastBit: interactively searching massive data," in *Journal of Physics: Conference Series*, vol. 180, no. 1. IOP Publishing, 2009, p. 012053.

[123] B. Dong, S. Byna, and K. Wu, "SDS: A framework for scientific data services," in *Proceedings of the 8th Parallel Data Storage Workshop*, ser. PDSW '13. New York, NY, USA: ACM, 2013, pp. 27–32.

[124] Y. Yao, B. P. Bowen, D. Baron, and D. Poznanski, "SciDB for high-performance array-structured science data at NERSC." *Computing in Science and Engineering*, vol. 17, no. 3, pp. 44–52, 2015.

[125] R. Gardner, S. Campana, G. Duckeck, J. Elmsheuser, A. Hanushevsky, F. G. Hönig, J. Iven, F. Legger, I. Vukotic, W. Yang, and the Atlas Collaboration, "Data federation strategies for ATLAS using XRootD," *Journal of Physics: Conference Series*, vol. 513, no. 4, p. 042049, 2014.

[126] G. Grider, "Exa-scale FSIO - Can we get there? Can we a ord to?" Presented at the 7th IEEE International Workshop on Storage Network Architecture and Parallel I/O, May 2011.

[127] N. Liu, J. Cope, P. Carns, C. Carothers, R. Ross, G. Grider, A. Crume, and C. Maltzahn, "On the role of burst bu ers in leadership-class storage systems," in *Proceedings of 28th IEEE MSST conference*, 2012.

[128] National Energy Research Scientific Computing Center, "Cori," https://www.nersc.gov/users/computational-systems/cori/, 2015.

[129] Cray Inc., "Cray XC40 Datawarp applications I/O accelerator," http://www.cray.com/sites/default/files/resources/CrayXC40-DataWarp.pdf, 2015.

[130] DataDirect Networks, "Technology Brief: Infinite Memory Engine," http://www.ddn.com/download/resource_library/solution_briefs/cloud_and_web_companies/DDN-IME-BurstBu er-TechnologyBrief.pdf, 2015.

[131] T. Jin, F. Zhang, Q. Sun, H. Bui, M. Romanus, N. Podhorszki, S. Klasky, H. Kolla, J. Chen, R. Hager *et al.*, "Exploring data staging across deep memory hierarchies for coupled data intensive simulation workflows," in *Parallel and Distributed Processing Symposium (IPDPS), 2015 IEEE International*. IEEE, 2015, pp. 1033–1042.

[132] Q. Sun, T. Jin, M. Romanus, H. Bui, F. Zhang, H. Yu, H. Kolla, S. Klasky, J. Chen, and M. Parashar, "Adaptive data placement for staging-based coupled scientific workflows," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. ACM, 2015, p. 65.

[133] J. Y. Choi, K. Wu, J. C. Wu, A. Sim, Q. G. Liu, M. Wolf, C. Chang, and S. Klasky, "Icee: Wide-area in transit data processing framework for near real-time scientific applications," *4th SC Workshop on Petascale (Big) Data Analytics: Challenges and Opportunities in conjunction with SC13*, 2013.

[134] M. F. Aktas, G. Haldeman, and M. Parashar, "Scheduling and flexible control of bandwidth and in-transit services for end-to-end application workflows," *Future Generation Computer Systems*, 2015.

[135] C. Docan, M. Parashar, and S. Klasky, "Dataspaces: an interaction and coordination framework for coupled simulation workflows," *Cluster Computing*, vol. 15, no. 2, pp. 163–181, 2012.

[136] E. Riedel, G. Gibson, and C. Faloutsos, "Active storage for large-scale data mining and multimedia applications," in *Proceedings of 24th Conference on Very Large Databases*, 1998, pp. 62–73.

[137] S. W. Son, S. Lang, P. Carns, R. Ross, R. Thakur, B. Ozisikyilmaz, P. Kumar, W.-K. Liao, and A. Choudhary, "Enabling active storage on parallel I/O software stacks," in *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on*, 2010, pp. 1–12.

[138] D. Reiner, G. Press, M. Lenaghan, D. Barta, and R. Urmston, "Information lifecycle management: the EMC perspective," in *Data Engineering, 2004. Proceedings. 20th International Conference on*, March 2004, pp. 804–807.

[139] Amazon Web Services Inc., "Amazon Glacier," https://aws.amazon.com/glacier/, 2015.

[140] D. N. Williams, R. Drach, R. Ananthakrishnan, I. Foster, D. Fraser, F. Siebenlist, D. Bernholdt, M. Chen, J. Schwidder, S. Bharathi *et al.*, "The Earth System Grid: Enabling access to multimodel climate simulation data," *Bulletin of the American Meteorological Society*, vol. 90, no. 2, pp. 195–205, 2009.

[141] R. Moore, "Towards a theory of digital preservation," *International Journal of Digital Curation*, vol. 3, no. 1, pp. 63–75, 2008.

[142] H. Kim and M. Parashar, "Cometcloud: An autonomic cloud engine," *Cloud Computing: Principles and Paradigms*, pp. 275–297, 2011.

[143] A. Shoshani and D. Rotem, Eds., *Scientific Data Management: Challenges, Technology, and Deployment*. Chapman & Hall/CRC Press, 2010.

[144] "Data crosscutting requirements review," http://science.energy.gov/~/media/ascr/pdf/program-documents/docs/ASCR_DataCrosscutting2_8_28_13.pdf, Tech. Rep., 2013, workshop report sponsored by US Department of Energy, Office of Science.

[145] E. Duval, W. Hodgins, S. Sutton, and S. L. Weibel, "Metadata principles and practicalities," *D-lib Magazine*, vol. 8, no. 4, p. 16, 2002.

[146] P. Buneman, S. Khanna, and T. Wang-Chiew, "Why and where: A characterization of data provenance," in *Database Theory – ICDT 2001*. Springer, 2001, pp. 316–330.

[147] Y. L. Simmhan, B. Plale, and D. Gannon, "A survey of data provenance in e-science," *SIGMOD Rec.*, vol. 34, no. 3, pp. 31–36, Sep. 2005. [Online]. Available: http://doi.acm.org/10.1145/1084805.1084812

[148] E. G. Stephan, P. Pinheiro da Silva, and K. Kleese van Dam, *Bridging the gap between scientific data producers and consumers: a provenance approach*. CRC Press, 2013, pp. 279–300.

[149] J. G. Frey, A. Milsted, D. Michaelides, and D. D. Roure, "Myexperimentalscience, extending the 'workflow'," *Concurrency and Computation: Practice and Experience*, vol. 25, no. 4, pp. 481–496, 2013.

[150] R. Brun and F. Rademakers, "Root – an object oriented data analysis framework," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 389, no. 1, pp. 81–86, 1997.

[151] "NetCDF climate and forecast (CF) metadata conventions," http://www.cgd.ucar.edu/cms/eaton/netcdf/CF-20010629.htm, 2001.

[152] B. Momjian, *PostgreSQL: introduction and concepts*. Addison-Wesley New York, 2001, vol. 192.

[153] D. Smiley, E. Pugh, K. Parisa, and M. Mitchell, *Apache Solr Enterprise Search Server*. Packt Publishing Ltd, 2015.

[154] B. Domenico, J. Caron, E. Davis, R. Kambic, and S. Nativi, "Thredds: Incorporating real-time environmental data and interactive analysis tools into nsdl," *J. Digital Inf*, vol. 2, no. 4, 2002.

[155] S. P. Callahan, J. Freire, E. Santos, C. E. Scheidegger, C. T. Silva, and H. T. Vo, "Vistrails: visualization meets data management," in *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*. ACM, 2006, pp. 745–747.

[156] E. Santos, J. Poco, Y. Wei, S. Liu, B. Cook, D. N. Williams, and C. T. Silva, "Uv-cdat: analyzing climate datasets from a user's perspective," *Computing in Science & Engineering*, vol. 15, no. 1, pp. 94–103, 2013.

[157] P. Missier, S. C. Dey, K. Belhajjame, V. Cuevas-Vicenttín, and B. Ludäscher, "D-prov: extending the prov provenance model with workflow structure," in *TaPP*, 2013. [Online]. Available: https://www.usenix.org/system/files/conference/tapp13/tapp13-final3.pdf

[158] L. Moreau, B. Cli ord, J. Freire, J. Futrelle, Y. Gil, P. Groth, N. Kwasnikowska, S. Miles, P. Missier, J. Myers *et al.*, "The open provenance model core specification (v1. 1)," *Future Generation Computer Systems*, vol. 27, no. 6, pp. 743–756, 2011.

[159] P. Missier, K. Belhajjame, and J. Cheney, "The w3c prov family of specifications for modelling provenance metadata," in *Proceedings of the 16th International Conference on Extending Database Technology*. ACM, 2013, pp. 773–776.

[160] D. D. Lamanna and A. Maccioni, "Renewable energy data sources in the semantic web with openwatt." in *EDBT/ICDT Workshops*, 2014, pp. 128–133.

[161] C. Tilmes, P. Fox, X.-L. Ma, D. L. McGuinness, A. P. Privette, A. Smith, A. Waple, S. Zednik, and J. G. Zheng, "Provenance representation for the national climate assessment in the global change information system," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 51, no. 11, pp. 5160–5168, 2013.

[162] J. C. Wright, M. Greenwald, J. Stillerman, G. Abla, B. Chanthavong, S. Flanagan, D. Schissel, X. Lee, A. Romosan, and A. Shoshani, "The mpo api: A tool for recording scientific workflows," *Fusion Engineering and Design*, vol. 89, no. 5, pp. 754–757, 2014.

[163] F. Costa, V. Silva, D. de Oliveira, K. A. Ocana, and M. Mattoso, "Towards supporting provenance gathering and querying in di erent database approaches," in *Provenance and Annotation of Data and Processes*, ser. Lecture Notes in Computer Science, B. Ludascher and B. Plale, Eds. Springer International Publishing, 2015, vol. 8628, pp. 254–257. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-16462-5_26

[164] P. Macko and M. Seltzer, "A general-purpose provenance library." in *TaPP*, 2012. [Online]. Available: https://www.usenix.org/system/files/conference/tapp12/tapp12-final9.pdf

[165] D. Ghoshal and B. Plale, "Provenance from log files: A bigdata problem," in *Proceedings of the Joint EDBT/ICDT 2013 Workshops*, ser. EDBT '13. ACM, 2013, pp. 290–297. [Online]. Available: http://doi.acm.org/10.1145/2457317.2457366

[166] A. Marinho, L. Murta, C. Werner, V. Braganholo, S. M. S. d. Cruz, E. Ogasawara, and M. Mattoso, "Provmanager: a provenance management system for scientific workflows," *Concurrency and Computation: Practice and Experience*, vol. 24, no. 13, pp. 1513–1530, 2012.

[167] L. Murta, V. Braganholo, F. Chirigati, D. Koop, and J. Freire, "noworkflow: Capturing and analyzing provenance of scripts," in *Provenance and Annotation of Data and Processes*, ser. Lecture Notes in

Computer Science, B. Ludascher and B. Plale, Eds. Springer, 2015, vol. 8628, pp. 71–83. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-16462-5_6

[168] I. Suriarachchi, Q. G. Zhou, and B. Plale, "Komadu: A capture and visualization system for scientific data provenance," *Journal of Open Research Software*, vol. 3, no. 1, Mar. 2015. [Online]. Available: http://openresearchsoftware.metajnl.com/article/view/jors.bq/

[169] K. K. van Dam, R. LaMothe, P. Sharma, D. Zarzhitsky, A. Vishnu, E. Stephan, W. Smith, T. Elsethagen, and M. Thomas, "Building the analysis in motion infrastructure," PNNL, Tech. Rep. PNNL-24340, 2015.

[170] Y.-W. Cheah, B. Plale, J. Kendall-Morwick, D. Leake, and L. Ramakrishnan, "A noisy 10gb provenance database," in *Business Process Management Workshops*, ser. Lecture Notes in Business Information Processing, F. Daniel, K. Barkaoui, and S. Dustdar, Eds. Springer Berlin Heidelberg, 2012, vol. 100, pp. 370–381.

[171] Y.-W. Cheah and B. Plale, "Provenance analysis: Towards quality provenance," in *E-Science 2012*. IEEE, Oct. 2012, pp. 1–8.

[172] Prabhat, O. Rübel, S. Byna, K. Wu, F. Li, M. Wehner, and E. W. Bethel, "TECA: A parallel toolkit for extreme climate analysis," in *Third Workshop on Data Mining in Earth System Science (DMESS 2012) at the International Conference on Computational Science (ICCS 2012)*, *Omaha*, *Nebraska*, Jun. 2012.

[173] J. Gallicchio and M. D. Schwartz, "Quark and gluon tagging at the LHC," *Physical Review Letters*, vol. 107, no. 17, Oct. 2011.

[174] R. Saurí, R. Knippen, M. Verhagen, and J. Pustejovsky, "Evita: A robust event recognizer for qa systems," in *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT '05)*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2005, pp. 700–707.

[175] A. Conesa, S. Gotz, J. M. Garcia-Gomez, J. Terol, M. Talon, and M. Robles, "Blast2go: a universal tool for annotation, visualization and analysis in functional genomics research," *Bioinformatics*, vol. 21, no. 18, pp. 3674–3676, Sep. 2005.

[176] G. Dennis Jr, B. T. Sherman, D. A. Hosack, J. Yang, W. Gao, H. C. Lane, R. A. Lempicki *et al.*, "DAVID: database for annotation, visualization, and integrated discovery," *Genome biol*, vol. 4, no. 5, p. P3, 2003.

[177] A. Kozomara and S. Griffiths-Jones, "miRBase: integrating microRNA annotation and deep-sequencing data," *Nucleic Acids Research*, vol. 39, no. Database, pp. D152–D157, Jan. 2011.

[178] A. Marchler-Bauer, S. Lu, J. B. Anderson, F. Chitsaz, M. K. Derbyshire, C. DeWeese-Scott, J. H. Fong, L. Y. Geer, R. C. Geer, N. R. Gonzales, M. Gwadz, D. I. Hurwitz, J. D. Jackson, Z. Ke, C. J. Lanczycki, F. Lu, G. H. Marchler, M. Mullokandov, M. V. Omelchenko, C. L. Robertson, J. S. Song, N. Thanki, R. A. Yamashita, D. Zhang, N. Zhang, C. Zheng, and S. H. Bryant, "CDD: a conserved domain database for the functional annotation of proteins," *Nucleic Acids Research*, vol. 39, no. Database, pp. D225–D229, Jan. 2011.

[179] T. Rogerson, D. J. Cai, A. Frank, Y. Sano, J. Shobe, M. F. Lopez-Aranda, and A. J. Silva, "Synaptic tagging during memory allocation," *Nature Reviews Neuroscience*, vol. 15, no. 3, pp. 157–169, Feb. 2014.

[180] K. Mavromatis, K. Chu, N. Ivanova, S. Hooper, V. Markowitz, and N. Kyrpides, "Gene context analysis in the integrated microbial genomes (IMG) data management system," *PLoS ONE*, vol. 4, no. 11, p. e7979, 2009.

[181] A. Romosan, A. Shoshani, K. Wu, V. Markowitz, and K. Mavrommatis, "Accelerating gene context analysis using bitmaps," in *Proceedings of the 25th International Conference on Scientific and Statistical Database Management*. ACM, 2013, p. 26.

[182] J. T. Leek and R. D. Peng, "Opinion: Reproducible research can still be wrong: Adopting a prevention approach," *Proceedings of the National Academy of Sciences*, vol. 112, no. 6, pp. 1645–1646, 2015. [Online]. Available: http://www.pnas.org/content/112/6/1645.short

[183] "Journals unite for reproducibility," pp. 7–7, Nov. 2014.

[184] V. Cuevas-Vicenttin, S. Dey, M. L. Y. Wang, T. Song, and B. Ludascher, "Modeling and querying scientific workflow provenance in the D-OPM." IEEE, Nov. 2012, pp. 119–128.

[185] D. Garijo and Y. Gil, "Towards open publication of reusable scientific workflows: Abstractions, standards and linked data," Project Report, 2012.

[186] T. Heinis and G. Alonso, "Efficient lineage tracking for scientific workflows." ACM Press, 2008, pp. 1007–1018.

[187] C. E. Scheidegger, H. T. Vo, D. Koop, J. Freire, and C. T. Silva, "Querying and reusing workflows with VsTrails." ACM Press, 2008, p. 1251. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1376616.1376747

[188] D. Chapp, T. Johnston, M. Becchi, and M. Taufer, "Reproducible numerical accuracy at the extreme scale," ASCR Workflow Workshop White Paper, April 2015, 2015.

[189] ——, "Numerical reproducibility challenges on extreme scale multi-threading gpus," http://on-demand.gputechconf.com/gtc/2015/presentation/S5245-Michela-Taufer.pdf, 2015, presented at GTC2015.

[190] T. Clark, "Next generation scientific publishing and the web of data," *Semantic Web J*, 2014. [Online]. Available: http://semantic-web-journal.org/system/files/swj670.pdf

[191] N. A. Vasilevsky, M. H. Brush, H. Paddock, L. Ponting, S. J. Tripathy, G. M. LaRocca, and M. A. Haendel, "On the reproducibility of science: unique identification of research resources in the biomedical literature," *PeerJ*, vol. 1, p. e148, Sep. 2013.

[192] "Digital curation centre." [Online]. Available: http://www.dcc.ac.uk/

[193] D. Williams and K. a. Kleese-Van Dam, *Working Group on Virtual Data Integration: A Report from the August 13?14, 2015, Workshop*. U.S. Department of Energy Office of Science. DOI:10.2172/1227017, 2016.

[194] Rocca-Serra and Philippe, "Specification documentation: release candidate 1, ISA-TAB 1.0," http://isatab.sourceforge.net/docs/ISA-TAB_release-candidate-1_v1.0_24nov08.pdf, last accessed March 2016.

[195] D. C. M. Initiative, "Dcmi specifications," http://dublincore.com/specifications/, last accessed March 2016.

[196] [Online]. Available: http://www.nexusformat.org

[197] G. Shipman, S. Campbell, D. Dillow, M. Doucet, J. Kohl, G. Granroth, R. Miller, D. Stansberry, T. Proffen, and R. Taylor, "Accelerating data acquisition, reduction, and analysis at the spallation neutron source," in *e-Science (e-Science), 2014 IEEE 10th International Conference on*, vol. 1. IEEE, 2014, pp. 223–230.

[198] [Online]. Available: https://monitor.sns.gov

[199] (2014). [Online]. Available: http://dx.doi.org/10.5286/SOFTWARE/ICAT

[200] O. Arnold, J.-C. Bilheux, J. Borreguero, A. Buts, S. I. Campbell, L. Chapon, M. Doucet, N. Draper, R. F. Leal, M. Gigg *et al.*, "Mantid—data analysis and visualization package for neutron scattering and μ sr experiments," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 764, pp. 156–166, 2014.

[201] [Online]. Available: http://horace.isis.rl.ac.uk/Main_Page

[202] R. T. Azuah, L. R. Kneller, Y. Qiu, P. L. Tregenna-Piggott, C. M. Brown, J. R. Copley, and R. M. Dimeo, "Dave: a comprehensive software suite for the reduction, visualization, and analysis of low energy neutron spectroscopic data," *Journal of Research of the National Institute of Standards and Technology*, vol. 114, no. 6, pp. 341–358, 2009.

[203] [Online]. Available: http://horace.isis.rl.ac.uk/Main_Page

[204] [Online]. Available: http://www.exelisvis.com/ProductsServices/IDL.aspx

[205] A. Henderson, J. Ahrens, and C. Law, *The ParaView Guide*. Kitware Clifton Park, NY, 2004.

[206] S. Chi, F. Ye, W. Bao, M. Fang, H. Wang, C. Dong, A. T. Savici, G. E. Granroth, M. B. Stone, and R. S. Fishman, "Neutron scattering study of spin dynamics in superconducting (tl, rb) 2 fe 4 se 5," *Physical Review B*, vol. 87, no. 10, p. 100501, 2013.

[207] [Online]. Available: https://aws.amazon.com

[208] [Online]. Available: https://azure.microsoft.com

[209] A. Larson and R. B. Von Dreele, "General structure analysis system (gsas)," Los Alamos National Laboratory, Tech. Rep. LAUR 86-748, 2000.

[210] J. Rodriguez-Carvajal, *Physica B.*, vol. 192, p. 55, 1993.

[211] B. Adams, L. Bauman, W. Bohnho , K. Dalbey, M. Ebeida, J. Eddy, M. Eldred, P. Hough, K. Hu, J. Jakeman, L. Swiler, and D. Vigil, "Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 5.4 user's manual," Sandia National Laboratory, Tech. Rep. SAND2010-2183, 2013.

[212] I. Bustinduy, F. Bermejo, T. Perring, and G. Bordel, "A multiresolution data visualization tool for applications in neutron time-of-flight spectroscopy," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 546, no. 3, pp. 498–508, 2005.

[213] ——, "Experimental neutron spectroscopy data visualization: Adaptive tessellation algorithm," *Review of scientific instruments*, vol. 78, no. 4, p. 043901, 2007.

[214] S. Pennycook and P. D. Nellist, *Scanning Transmission Electron Microscopy: Imaging and Analysis*. New York: Springer, 2011.

[215] S. J. Pennycook, M. F. Chisholm, A. R. Lupini, M. Varela, K. van Benthem, A. Y. Borisevich, M. P. Oxley, W. Luo, and S. T. Pantelides, "Materials applications of aberration-corrected scanning transmission electron microscopy," in *Advances in Imaging and Electron Physics*, P. W. Hawkes, Ed., vol. 153. San Diego: Elsevier Academic Press Inc., 2008, pp. 327–+.

[216] C. Mody, *Instrumental Community: Probe Microscopy and the Path to Nanotechnology*. MIT Press, 2011.

[217] A. B. Yankovich, B. Berkels, W. Dahmen, P. Binev, S. I. Sanchez, S. A. Bradley, A. Li, I. Szlufarska, and P. M. Voyles, "Picometre-precision analysis of scanning transmission electron microscopy images of platinum nanocatalysts," *Nature Communications*, 2014.

[218] Y. M. Kim, J. He, M. D. Biegalski, H. Ambaye, V. Lauter, H. M. Christen, S. T. Pantelides, S. J. Pennycook, S. V. Kalinin, and A. Y. Borisevich, "Probing oxygen vacancy concentration and homogeneity in solid-oxide fuel-cell cathode materials on the subunit-cell level," *Nature Materials*, vol. 11, pp. 888–894, 2012.

[219] Y. M. Kim, A. Morozovska, E. Eliseev, M. P. Oxley, R. Mishra, S. M. Selbach, T. Grande, S. T. Pantelides, S. V. Kalinin, and A. Y. Borisevich, "Direct observation of ferroelectric field e ect and vacancy-controlled screening at the bifeo3/laxsr1-xmno3 interface," *Nature Materials*, vol. 13, pp. 1019–1025, 2014.

[220] H. J. Chang, S. V. Kalinin, A. N. Morozovska, M. Huijben, Y. H. Chu, P. Yu, R. Ramesh, E. A. Eliseev, G. S. Svechnikov, S. J. Pennycook, and et al., "Atomically resolved mapping of polarization and electric fields across ferroelectric/oxide interfaces by z-contrast imaging," *Advanced Materials*, vol. 23, pp. 2474–+, 2011.

[221] C. T. Nelson, B. Winchester, Y. Zhang, S. J. Kim, A. Melville, C. Adamo, C. M. Folkman, S. H. Baek, C. B. Eom, D. G. Schlom, and et al., "Spontaneous vortex nanodomain arrays at ferroelectric heterointerfaces," *Nano Letters*, vol. 11, pp. 828–834, 2011.

[222] C. L. Jia, V. Nagarajan, J. Q. He, L. Houben, T. Zhao, R. Ramesh, K. Urban, and R. Waser, "Unit-cell scale mapping of ferroelectricity and tetragonality in epitaxial ultrathin ferroelectric films," *Nature Materials*, vol. 6, pp. 64–69, 2007.

[223] C. L. Jia, K. W. Urban, M. Alexe, D. Hesse, and I. Vrejoiu, "Direct observation of continuous electric dipole rotation in flux-closure domains in ferroelectric pb(zr,ti)o(3)," *Science*, vol. 331, pp. 1420–1423, 2011.

[224] C. L. Jia, S. B. Mi, M. Faley, U. Poppe, J. Schubert, and K. Urban, "Oxygen octahedron reconstruction in the srtio(3)/laalo(3) heterointerfaces investigated using aberration-corrected ultrahigh-resolution transmission electron microscopy," *Physical Review B*, vol. 79, 2009.

[225] Y. M. Kim, A. Kumar, A. Hatt, A. N. Morozovska, A. Tselev, M. D. Biegalski, I. Ivanov, E. A. Eliseev, S. J. Pennycook, J. M. Rondinelli, and et al., "Interplay of octahedral tilts and polar order in bifeo3 films," *Advanced Materials*, vol. 25, pp. 2497–2504, 2013.

[226] A. Y. Borisevich, H. J. Chang, M. Huijben, S. Oxley, M. P.and Okamoto, M. K. Niranjan, J. D. Burton, E. Y. Tsymbal, Y. H. Chu, P. Yu, and et al., "Suppression of octahedral tilts and associated changes in electronic properties at epitaxial oxide heterostructure interfaces," *Physical Review Letters*, vol. 105, 2010.

[227] J. He, A. Borisevich, S. V. Kalinin, S. J. Pennycook, and S. T. Pantelides, "Control of octahedral tilts and magnetic properties of perovskite oxide heterostructures by substrate symmetry," *Physical Review Letters*, vol. 105, 2010.

[228] A. Borisevich, O. S. Ovchinnikov, H. J. Chang, M. P. Oxley, P. Yu, J. Seidel, E. A. Eliseev, A. N. Morozovska, R. Ramesh, S. J. Pennycook, and et al., "Mapping octahedral tilts and polarization across a domain wall in bifeo(3) from z-contrast scanning transmission electron microscopy image atomic column shape analysis," *Acs Nano*, vol. 4, pp. 6071–6079, 2010.

[229] H. J. Chang, S. V. Kalinin, S. Yang, P. Yu, S. Bhattacharya, P. P. Wu, N. Balke, S. Jesse, L. Q. Chen, R. Ramesh, and et al., "Watching domains grow: In-situ studies of polarization switching by combined scanning probe and scanning transmission electron microscopy," *Journal of Applied Physics*, vol. 110, 2011.

[230] C. T. Nelson, P. Gao, J. R. Jokisaari, C. Heikes, C. Adamo, A. Melville, S. H. Baek, C. M. Folkman, B. Winchester, Y. J. Gu, and et al., "Domain dynamics during ferroelectric switching," *Science*, vol. 334, pp. 968–971, 2011.

[231] A. Belianinov, V. Iberi, A. Tselev, M. A. Susner, M. A. McGuire, D. Joy, S. Jesse, A. J. Rondinone, S. V. Kalinin, and O. S. Ovchinnikova, "Polarization control via he-ion beam induced nanofabrication in layered ferroelectric semiconductors," *ACS Nano*, 2015, under Review.

[232] E. Lingerfelt, S. Jesse, A. Belianinov, E. Endeve, O. Ovchinnikov, M. B. Okatan, C. Symons, M. Shankar, and R. Archibald, "Near real-time scalable analysis of multi-dimensional nanophase materials imaging data with beam," http://sc15.supercomputing.org/, 2015.

[233] S. Jesse, R. K. Vasudevan, L. Collins, E. Strelcov, M. B. Okatan, A. Belianinov, A. P. Baddorf, R. Proksch, and S. V. Kalinin, "Band excitation in scanning probe microscopy: Recognition and functional imaging," vol. 65, pp. 519–536, 2014.

[234] A. Belianinov, Q. He, M. Kravchenko, S. Jesse, A. Borisevich, and S. V. Kalinin, "Identification of phases, symmetries and defects through local crystallography," *Nature communications*, vol. 6, 2015.

[235] J. S., M. Chi, A. Belianinov, S. V. Kalinin, A. Borisevich, and A. Lupini, "Big data analytics in scanning transmission electron microscopy ptychography," 2015, in preparation.

[236] A. Belianinov, R. Vasudevan, E. Strelcov, C. Steed, S. M. Yang, A. Tselev, S. Jesse, M. Biegalski, G. Shipman, and C. Symons, "Big data and deep data in scanning and electron microscopies: Deriving functionality from multidimensional data sets," *Advanced Structural and Chemical Imaging*, vol. 1, pp. 1–25, 2015.

[237] A. Belianinov, S. V. Kalinin, and S. Jesse, "Complete information acquisition in dynamic force microscopy," *Nature communications*, vol. 6, 2015.

[238] W. Hoppe, "Beugung im inhomogenen primärstrahlwellenfeld i. prinzip einer phasenmessung von elektronenbeungungsinterferenzen," *Acta Crystallographica Section A: Crystal Physics, Di raction, Theoretical and General Crystallography*, vol. 25, pp. 495–501, 1969.

[239] D. Bourassa, S.-C. Gleber, S. Vogt, H. Yi, F. Will, H. Richter, C. H. Shin, and C. J. Fahrni, "3d imaging of transition metals in the zebrafish embryo by x-ray fluorescence microtomography," *Metallomics*, vol. 6, no. 9, pp. 1648–1655, 2014.

[240] R. Hegerl and W. Hoppe, "Influence of electron noise on three-dimensional image reconstruction," *Zeitschrift für Naturforschung A*, vol. 31, no. 12, pp. 1717–1721, 1976.

[241] B. F. McEwen, K. H. Downing, and R. M. Glaeser, "The relevance of dose-fractionation in tomography of radiation-sensitive specimens," *Ultramicroscopy*, vol. 60, no. 3, pp. 357–373, 1995.

[242] J. Deng, D. J. Vine, S. Chen, Y. S. Nashed, Q. Jin, N. W. Phillips, T. Peterka, R. Ross, S. Vogt, and C. J. Jacobsen, "Simultaneous cryo x-ray ptychographic and fluorescence microscopy of green algae," *Proceedings of the National Academy of Sciences*, vol. 112, no. 8, pp. 2314–2319, 2015.

[243] J. Deng, Y. S. Nashed, S. Chen, N. W. Phillips, T. Peterka, R. Ross, S. Vogt, C. Jacobsen, and D. J. Vine, "Continuous motion scan ptychography: characterization for increased speed in coherent x-ray imaging," *Optics express*, vol. 23, no. 5, pp. 5438–5451, 2015.

[244] Y. S. Nashed, D. J. Vine, T. Peterka, J. Deng, R. Ross, and C. Jacobsen, "Parallel ptychographic reconstruction," *Optics express*, vol. 22, no. 26, pp. 32 082–32 097, 2014.

[245] J. Gibbs, K. Mohan, E. Gulsoy, A. Shahani, X. Xiao, C. Bouman, M. De Graef, and P. Voorhees, "The three-dimensional morphology of growing dendrites," *Scientific reports*, vol. 5, 2015.

[246] T. Bicer, D. Gursoy, R. Kettimuthu, F. De Carlo, G. Agrawal, and I. T. Foster, "Rapid tomographic image reconstruction via large-scale parallelization," in *Euro-Par 2015: Parallel Processing*. Springer, 2015, pp. 289–302.

[247] B. H. Toby, D. Gürsoy, F. De Carlo, N. Schwarz, H. Sharma, and C. J. Jacobsen, "Practices and standards for data and processing at the APS," *Synchrotron Radiation News*, vol. 28, no. 2, pp. 15–21, 2015.

[248] I. Foster, R. Ananthakrishnan, B. Blaiszik, K. Chard, R. Osborn, S. Tuecke, M. Wilde, and J. Wozniak, "Networking materials data: Accelerating discovery at an experimental facility," in *Big Data and High Performance Computing*, L. Grandinetti and G. Joubert, Eds., In press, 2015.

[249] J. M. Wozniak, H. Sharma, T. G. Armstrong, M. Wilde, J. D. Almer, and I. Foster, "Big data staging with MPI-IO for interactive x-ray science," in *Proceedings of the 2014 IEEE/ACM International Symposium on Big Data Computing*. IEEE Computer Society, 2014, pp. 26–34.

[250] M. Wilde, M. Hategan, J. M. Wozniak, B. Cli ord, D. S. Katz, and I. Foster, "Swift: A language for distributed parallel scripting," *Parallel Computing*, vol. 37, no. 9, pp. 633–652, 2011.

[251] D. Gürsoy, F. De Carlo, X. Xiao, and C. Jacobsen, "TomoPy: a framework for the analysis of synchrotron tomographic data," *Journal of synchrotron radiation*, vol. 21, no. 5, pp. 1188–1193, 2014.

[252] B. H. Toby, "EXPGUI, a graphical user interface for GSAS," *Journal of applied crystallography*, vol. 34, no. 2, pp. 210–213, 2001.

[253] [Online]. Available: http://arxiv.org/abs/1307.7335

[254] [Online]. Available: http://aliceinfo.cern.ch/Public/en/Chapter2/Chap2_TPC.html

[255] M. A. T. J. S. Marshall, "The pandora software development kit for particle flow calorimetry," *Journal of Physics: Conference Series*, no. 396, 2012.

[256] "Building for Discovery: Strategic Plan for U.S. Particle Physics in the Global Context, Report of the Particle Physics Project Prioritization Panel (P5)," http://www.usparticlephysics.org/p5/.

[257] http://www.darkenergysurvey.org/.

[258] e. a. M. Levi, arXiv:1308.0847 [astro-ph.CO].

[259] e. a. P.A. Abell, arXiv:0912.0201 [astro-ph.IM].

[260] e. a. K.N. Abazajian, arXiv:1309.5381 [astro-ph.CO].

[261] https://www.skatelescope.org/.

[262] http://lz.lbl.gov/.

[263] http://cdms.berkeley.edu/.

[264] http://fgst.slac.stanford.edu/.

[265] http://www.hawc-observatory.org/.

[266] http://www.sdss.org/.

[267] D. Petravick, "Case study in the ASCR/HEP Exascale Requirements Review report."

[268] Talk by J. Kantor on LSST data management, http://www.lsst.org/sites/default/files/documents/DMIntroduction-Kantor.pdf.

[269] "DESI Conceptual Design Report, Section 6," http://desi.lbl.gov/wp-content/uploads/2014/04/DESI_CDR_20140827_1135.pdf.

[270] Talk by J. Borrill, http://www.nersc.gov/assets/Uploads/CMB-Borrill.pdf.

[271] A. Connolly, S. Habib, A. Szalay, and et al., "Snowmass 2013 Study Electronic Proceedings," https://www-public.slac.stanford.edu/snowmass2013/Index.aspx; computationally relevant Snowmass documents are collected at http://hepfce.org/documents/, arXiv:1311.2841 [astro-ph.CO].

[272] S. Habib, R. Roser, C. Tull, B. Hendrickson, R. Ross, and A. Shoshani, http://science.energy.gov/$\sim$/media/ascr/pdf/program-documents/docs/\HEP_ASCR_Data_Summit_Report_April_2013.pdf.

[273] The report will be available soon at http://hepfce.org/.

[274] http://hepfce.org/working-groups/.

Appendices

Appendix: 22

Workshop Process and Agenda

22.1 Before the Workshop: Gathering Data

Prior to the workshop, the EOS representatives prepared science use cases that speak to several di erent specific issues related to the overall workshop theme. These topics include:

- Present- or near-term issues. We requested a description of the science facility, how the facility or experiment "does science" with the EOD they collect, a "flowchart" (verbal or pictorial) describing the data lifecycle starting with data acquisition and including all processing stages and going through dissemination.
- Future issues. We requested information from the same categories as for the present- or near-term issues.
- Data lifecycle. We requested information about how data is used and key issues throughout the data lifecycle at each of the primary data lifecycle stages in the present and future views.
- Data requirements. For each stage in the data lifecycle, we requested information about data "speeds and feeds," throughput requirements, and specific data-centric capabilities needed for the specific science use case.
- Impediments, gaps, needs, challenges. We asked for 3–5 data-centric impediments or barriers facing each science project facing them now or going into the future.

We provided each EOS representative with a LaTeX template, which they used in preparing their science use case narrative. We distributed PDF versions of the science use cases to attendees prior to the workshop.

22.2 At the Workshop: Identifying Themes

At the workshop, we organized the agenda in a way so as to facilitate focused discussions around the science use cases and to facilitate dialogue between EOS and math/computer science representatives. We organized the set of EOS presentations roughly by program office: BER, BES, and HEP, which resulted in approximately four presentations in each category.

The pattern we used at the workshop is as follows:

- For each of the BER, BES, and HEP EOS project groups:

  - Science use case presentations (plenary). Each EOS representative gave an oral presentation describing their use case.
  - Focused discussion (breakouts). We had four breakout groups, each of which consisted of one or two EOS representatives and approximately one fourth of the math and computer science attendees. Each focused discussion group lasted about 15–20 minutes. Then, we would rotate the EOS representatives to the next group. This way, we facilitated in-depth discussion and question-answer about all of the science cases.
  - Lighting presentations (plenary). Next, an "area lead" for each major math or computer science group (roughly speaking, the major themes, which the workshop organizers determined beforehand) would prepare a brief presentation that listed the top N research challenges for their area that would need progress in order to meet science objectives.
  - Science feedback (plenary). We gave the EOS representatives the opportunity to provide immediate feedback so that the math and computer science representatives could "fine tune" their assessments.

- Report drafting. After repeating the above process three times, once for each of BES, HEP, and BER group of EOS projects, each of the math and computer science areas had in-hand a list of the major research themes for their area. Also at this point, this list of themes had undergone a degree of vetting by the EOS representatives. The math and computer science attendees were given the opportunity to begin drafting their report sections. The drafting process included use of a LaTeX template and a specific writing formula intended to promote some level of consistency across the di erent sections.

22.3 After the Workshop: Community Input

In the weeks after the workshop, we engaged in a period of report writing, in which ideas were refined. The workshop organizers iterated with each of the area leads in a process aimed at clarifying ideas, improving the exposition, and so forth.

As part of the report-writing process, we reached out into the math and computer science community to request community input on the report. Specifically, we solicited comments about major thematic areas we may have missed. In this process, we reached out to approximately 80 persons who are math or computer science subject matter experts. We revised the workshop report in response to their input, and have acknowledged their helpful input (24.5).

Appendix: 23

Data Growth Rates from EOS Projects

BER Use Cases

Data management, Analysis and Dissemination at EMSL(§11)

Processing stage	Present/Near term	Long term
Data acquisition rate: maximum rate(s) and monthly or annual totals	100Mbps maximum data rate; 3.6 TB monthly	1 GB/s maximum data rate; 18 TB monthly
Experiment-side processing	data reduction, preliminary analysis	data reduction, metadata collection, collaborative analysis
Real-time constraints, turnaround time from collection to result for experimental control	3 d	1 h
Metadata/provenance capture	Metadata from instruments and automated data processing	Fully automated metadata and provenance capture together with metadata from experimental protocols and sample generation

Climate Simulation and Analysis (§12)

Processing stage	Present/Near term	Long term
Data rate for production simulations on LCF *standard output* : maximum rate(s) and annual total assuming we could do this every day of the year (we do not). *no attempt to optimize strategies to archive data (e.g. compression)*	5 TB/day maximum data rate; 1.5 PB annually	Assume increase by factor of 50 over present day: 250 TB/day maximum data rate; 50 PB annually

Current strategies assume data published at the rates above will be published (shared across networks routinely)	We don't do this now, but are trying: Data is currently written to HPSS	We do not do this now, but wish we could
Stage 1 processing: data reduction of LCF output, preliminary analysis	(input 5 TB/day → 50 GB/day output	250 TB/day → 5 TB/day output
Stage 2 processing: Routine visualization and analysis of multiple data sets output from Stage 1 above	multiple 50 GB data sets daily	multiple 5 TB data sets daily.

Atmospheric Radiation Measurement Climate Research Facility (§13)

Processing stage	Present/Near term	Long term
Data acquisition rate: monthly or annual totals	18 TB/mo.	5 PB observational data annually, 1 PB model data annually
Network data transfer rates by site	SGP 100 Mbps, anticipated increase to 1 GB/s in FY16; most mobile facility sites around 1–2 Mbps satellite link; Antarctica bandwidth limited to 512 kbps	May remain the same
Archive download rate	Currently about 10–15 TB per month, most individual data orders less than 100 GB	Expected to be much higher; individual LES model download rates may be of a few TB
Example new large data stream: radar Doppler spectra at Azores site	Five months of spectra data had volumes: 6.9 TB, 7.3 TB, 737 GB, 4.7 TB, 11 TB	This data stream will soon be collected regularly at all 5–6 sites
Searching, merging, and subsetting data	Data discovery tool can search and subset by variable, site, and measurement and will soon be able to merge some data streams into a common time using ADI	Searching and subsetting by atmospheric state, cloud type, etc.

BES Use Cases

Advanced Light Source (§14)

Processing stage	Present/Near term	Long term
Data acquisition rate: maximum rate(s) and monthly or annual totals	10 Gbps max; 140 TB/month average (raw data)	80 Gbps max; 1.5 PB/month (raw data)
Experiment-side processing	data reduction, tomographic reconstruction, etc.	In addition, add higher-level feature extraction/identification to guide experimental system

Real-time constraints, turnaround time from collection to result for experimental control	varies among 40 beamlines from sub-second to minutes	varies by beamline, increasing numbers of beamlines will need sub-second feedback
Metadata/provenance capture	Varies by beamline	Coordinated system for capturing metadata and data from all beamlines

Linac Coherent Light Source (§15)

Processing stage	Present/Near term	Long term
Data acquisition rate: maximum rate(s) and monthly or annual totals	1-10 GB/s maximum data rate; 1.5 PB annually	100 GB/s maximum data rate; 15 PB annually
Experiment-side processing	Detector calibration, feature extraction, histogramming, visualization	Detector calibration, feature extraction, histogramming, visualization
Real-time constraints, turnaround time from collection to result for experimental control	1–10 sec	1–10 sec

Use Case for Data at the Oak Ridge National Laboratory Neutron Sources (§16)

Processing stage	Present/Near term	Long term
Data acquisition rate: maximum rate(s) and monthly or annual totals	500 MB/s maximum data rate; 0.3 PB annually	5 GB/s maximum data rate; 1 PB annually
Experiment-side processing	data reduction, traditional analysis	data reduction, traditional analysis, electronic notebook meta data
Real-time constraints, turnaround time from collection to result for experimental control	5–15 minutes	5 sec
Metadata/provenance capture	This is done for most automated experiment variables and for reduction	capture appropriate analysis meta data and notebook style information.

Data and Analysis Requirements in Scanning Probe and Electron Microscopies (§17)

Processing stage	Present/Near term	Long term
Data acquisition rate: maximum rate(s) and monthly or annual totals	~10–100 GB/day for movies, ptychographic data sets, and Gmode SPM	~10 Mb/s for SPM, ~(1–10) GB/s for STEM in the full information capture modes

Experiment-side processing	Lossless compression, exploratory data analysis/multivariate statistics, deconvolution, feature extraction, pan-sharpening, compressed sensing, image registration	data reduction, metadata collection, collaborative analysis, real time theory feedback
Real-time constraints, turnaround time from collection to result for experimental control	In most cases offline in the day-month interval for analytics, minutes-hours for microscope operation	Real-time analytics (unmixing, atom finding, structure extraction) at imaging rates
Metadata/provenance capture	Metadata from instrument and environmental parameters. Storage of data analysis pathways	Capture appropriate analysis meta data and notebook style information. Cross-correlation of metadata with literature/web/data base searches

Computing within the APS for Data Collection and Analysis (§18)

Processing stage	Present/Near term	Long term
Data acquisition rate: maximum rate(s) and monthly or annual totals	Approximately 5 TB/day maximum burst rate for some beamlines; approximately 2 PB raw data annually across all beamlines	Approximately 1 PB/day maximum burst rate; 500 PB to 1,000 PB annually across all beamlines
Experiment-side processing	Over 30 facility-developed software packages available for imaging, spectroscopy, and scattering reduction, reconstruction, and analysis; simulations and modeling performed independent of data acquisition; near real-time analysis for MX and XPCS beamlines; most experiment collect data blindly with little or no acquisition time feedback	High-performance and distributed computing enabled software for routine reduction, reconstruction, analysis, and visualization; first-pass reduction, reconstruction, and analysis during data collection at most beamlines; simulations and modeling coupled with data acquisition for experiment steering
Real-time constraints, turnaround time from collection to result for experimental control	Near real-time processing available for XPCS	Technique dependent; real-time processing needed within ms or less for time-dependent techniques, such as XPCS, and within seconds for dynamic imaging; order of 1 minute or less required for static imaging and some di raction techniques
Metadata/provenance capture	Heterogeneous solutions dependent on beamline community and technique; automated collection for many MX and XPCS beamlines	Facility-wide tools and infrastructure for electronic logbooks, and automated metadata and provenance collection

HEP Use Cases

Data Challenges in the DUNE Experiment (§19)

Summary information not available. Please refer to the use case itself.

Open Numerical Laboratories (§20)

Processing stage	Present/Near term	Long term
Data acquisition rate: maximum rate(s) and monthly or annual totals	10 GB/s maximum data rate; 500–800 TB annually	100 GB/s maximum data rate; 10–20 PB annually
Experiment-side processing	some in-situ analysis, some near-line posterior analysis	in-situ triggers, posterior community analysis using immersive tools, comparisons to experimental data
Real-time constraints, turn around time from collection to result for experimental control	N/A today	15 mins for large-scale interactive analyses, 6 hours for batch jobs
Metadata/provenance capture	Limited info in file headers	Fully automated metadata and provenance and capture, stored in easy to search databases

DOE HEP Cosmic Frontier Use Cases (§21)

Summary information not available. Please refer to the use case itself.

Appendix: 24

Participants

24.1 Workshop Organizers

Bethel	Wes	LBNL	Organizer
Greenwald	Martin	MIT	Co-organizer
Lucy	Nowell	DOE/ASCR	Sponsor

24.2 Writing Leads

Bethel	Wes	LBNL	(Entire report)
Greenwald	Martin	MIT	(Entire report)
Ahrens	Jim	LANL	Section 4
Bhimji	Wahid	LBNL	Section 6
Buluç	Aydin	LBNL	Sections 2, 4
Carns	Phil	ANL	Section 8
Ferrier	Nicola	University of Chicago	Section 7
Foster	Ian	University of Chicago	Section 1
Geveci	Berk	Kitware, Inc.	Section 3
Hester	Mary	LBNL	Copyediting
Johnston	Bill	LBNL	Section 6
Jones	Terry	ORNL	Section 5
Klasky	Scott	ORNL	Section 1
Kleese van Dam	Kirsten	BNL	Sections 9, 10
Parashar	Manish	Rutgers University	Section 8
Ramakrishnan	Lavanya	LBNL	Section 7
Rogers	David	LANL	Section 4
Vetter	Je	Georgia Institute of Technology	Section 5
Wild	Stefan	ANL	Section 2
Wolf	Matthew	Georgia Institute of Technology	Section 3
Wiley	Steve	PNNL	Section 6
Wu	John	LBNL	Section 9

24.3 Science Use Cases

Granoth	Garrett	ORNL	Section 16
Habib	Salman	ANL	Section 21
Kalinin	Sergei	ORNL	Section 17
Parazzo	Amadeo	SLAC	Section 15
Parkinson	Dula	LBNL	Section 14
Potekhin	Maxim	BNL	Section 19
Rasch	Phil	PNNL	Section 12
Riihimaki	Laura	PNNL	Section 13
Sivaraman	Chitra	PNNL	Section 13
Szalay	Alex	Johns Hopkins University	Section 20
Toby	Brian	ANL	Section 18
Wiley	Steve	PNNL	Section 11

24.4 Workshop Attendees

Ahrens	Jim	LANL
Bethel	Wes	LBNL
Bhimji	Wahid	LBNL
Buluç	Aydin	LBNL
Carns	Phil	ANL
Childs	Hank	University of Oregon
Deelman	Ewa	ISI
Feng	Wuchun	Virginia Institute of Technology
Ferrier	Nicola	University of Chicago
Foster	Ian	University of Chicago
Gaither	Kelly	University of Texas–Austin, TACC
Geveci	Berk	Kitware, Inc.
Granoth	Garrett	ORNL
Greenwald	Martin	MIT
Habib	Salman	ANL
Hansen	Chuck	University of Utah
Hey	Tony	University of Washington
Johnston	Bill	LBNL
Jones	Terry	ORNL
Joy	Ken	University of California – Davis
Kalinin	Sergei	ORNL
Kamath	Chandrika	LLNL
Klasky	Scott	ORNL
Kleese van Dam	Kirsten	BNL
Moreland	Ken	SNL-NM
Parashar	Manish	Rutgers University
Parazzo	Amadeo	SLAC
Parkinson	Dula	LBNL
Peterka	Tom	University of Chicago
Potekhin	Maxim	BNL
Ramakrishnan	Lavanya	LBNL
Rasch	Phil	PNNL

Riihimaki	Laura	PNNL
Roser	Rob	FNAL
Samsel	Francesca	University of Texas–Austin
Sanderson	Allen	University of Utah
Shankar	Arjun	ORNL
Shen	Han-wei	Ohio State University
Sivaraman	Chitra	PNNL
Szalay	Alex	Johns Hopkins University
Toby	Brian	ANL
Uram	Tom	ANL
Ushizima	Daniela	LBNL
Vetter	Je	Georgia Institute of Technology
Ware	Colin	University of New Hampshire
Wild	Stefan	ANL
Wiley	Steve	PNNL
Williams	Dean	LLNL
Wolf	Matthew	Georgia Institute of Technology
Wu	John	LBNL

24.5 Community Input

Cappello	Franck	ANL
Constantinescu	Emil	ANL
Gruchalla	Ken	NREL
Ley er	Sven	ANL
Palanisamy	Giri	ORNL
Ross	Rob	ANL
Rübel	Oliver	LBNL
Shipman	Galen	LANL
Weber	Gunther	LBNL

Appendix: 25

Glossary of Common Acronyms

ACME	Accelerated Climate Modeling for Energy
AFM	Atomic force microscopy
ALCF	Argonne Leadership Computing Facility
ALICE	A Large Ion Collider Experiment
ALS	Advanced Light Source
AMI	Analysis and Monitoring Interface
AMO	Atomic, Molecular and Optical Science
API	Application programming interface
ARM	Atmospheric Radiation Measurement
ASCR	Advanced Scientific Computing Research
APS	Advanced Photon Source
BE PFM	Band excitation piezoresponse force microscopy
BEAM	Bellerophon Environment for Analysis of Materials
BEPS	Band excitation piezoresponse spectroscopy
BER	Biological and Environmental Research
CADES	Compute And Data Environment for Science
CAMERA	Center for Advanced Mathematics for Energy Research Applications
CCD	Charge Coupled Device
CMB	Cosmic Microwave Background
CMOS	complementary metal-oxide semiconductor
CNMS	Center for Nanophase Materials Sciences
CSI	Computational Science Initiative
CXI	Coherent X-Ray Imaging
DAQ	Data acquisition system
DISC	Data-Intensive Scalable Computing
DFT	Density Functional Theory
DMF	Data Management Facility
DMS	Data management system
DOE	Department of Energy
DUNE	Deep Underground Neutrino Experiment
EMSL	Environmental Molecular Sciences Laboratory
EO	Experimental and observational
EOD	Experimental and observational data
EOS	Experimental and observational science
EPICS	Experimental Physics and Industrial Control System

ESnet	Energy Sciences Network
FEL	Free electron laser
FIB	Focused ion beam
FNAL	Fermi National Accelerator Laboratory
FORC	First-order reversal curve
FORC IV	First-order reversal curves in current voltage measurements
FTP	File transfer protocol
G-PFM	General mode Piezoresponse Force Microscopy
GB/s	Gigabytes per second
Gbps	Gigabits per second
HFIR	High Flux Isotope Reactor
HPC	High performance computing
HTC	High Throughput Computing
I/O	Input/output
IFIM	Institute for Functional Imaging of Materials
ISA-Tab	Investigation, Study, and Assay Tabular Format
ISD	Interfacial shape distributions
LArTPC	Liquid Argon Time Projection Chamber
LANL	Los Alamos National Laboratory
LES	Large Eddy Scale
LHC	Large Hadron Collider
kbps	kilobits per second
MEC	Matter Extreme Conditions
MB/s	Megabytes per second
Mbps	Megabits per second
MD	Molecular dynamics
MG-RAST	Metagenomics Rapid Annotation using Subsystem Technology
MPI	Message passing interface
NASA	National Aeronautics and Space Administration
NCSA	National Center for Supercomputing Applications
NERSC	National Energy Research Scientific Computing Center
NMR	Nuclear magnetic resonance
OLCF	Oak Ridge Leadership Computing Facility
OPV	Organic photovoltaic
ORNL	Oak Ridge National Laboratory
PAD	Pixel Array Detector
PCDS	Photon Controls and Data Systems
PDFs	Probability density functions
PFS	Parallel file system
PI	Principal Investigator
QA	Quality Assurance
RHIC	Relativistic Heavy Ion Collider
SASE	Self-amplified spontaneous emission
SAXS	Small-angle X-ray scattering
SC	Office of Science
SEM	Scanning electron microscopy
SNS	Spallation Neutron Source
SPM	Scanning probe microscopy
SPS	Super Proton Synchrotron
(S)TEM	(Scanning) transmission electron microscopy
SXR	Soft X-Ray

TB/s	Terabytes per second
Tbps	Terabits per second
Tr-KPFM	Time resolved Kelvin probe force microscopy
USID	unique sample IDs
UQ	Uncertainty quantification
UX	User experience
V&V	Verification and validation
WAXS	Wide-angle X-ray scattering
XCS	X-ray Correlation Spectroscopy
XPP	X-ray Pump Probe
ZS	Zero suppression