SANDIA REPORT

SAND2023-08642
Printed August 2023

Sandia National Laboratories

Assessment of Data-Management Infrastructure Needs for Production Use of Advanced Machine Learning and Artificial Intelligence

Tri-Lab Level II Milestone (8554)

Benjamin Allan and Ron Oldfield
Sandia National Laboratories
{baallan,raoldfi}@sandia.gov

Charles Doutriaux and Katie Lewis
Lawrence Livermore National Laboratory
{doutriaux1,lewis66}@llnl.gov

James Ahrens, Benjamin Sims, Christine Sweeney, Divya Banesh, Quincy Wofford
Los Alamos National Laboratory
{ahrens,bsims,cahrens,dbanesh,qwofford}@lanl.gov

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185
Livermore, California 94550

Issued by Sandia National Laboratories, operated for the United States Department of Energy by National Technology & Engineering Solutions of Sandia, LLC.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from

U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone:	(865) 576-8401
Facsimile:	(865) 576-5728
E-Mail:	reports@osti.gov
Online ordering:	http://www.osti.gov/scitech

Available to the public from

U.S. Department of Commerce
National Technical Information Service
5301 Shawnee Road
Alexandria, VA 22312

Telephone:	(800) 553-6847
Facsimile:	(703) 605-6900
E-Mail:	orders@ntis.gov
Online order:	https://classic.ntis.gov/help/order-methods

Assessment of Data-Management Infrastructure Needs for Production Use of Advanced Machine Learning and Artificial Intelligence

Tri-Lab Level II Milestone (8554)

Benjamin Allan and Ron Oldfield
Sandia National Laboratories
{baallan,raoldfi}@sandia.gov

Charles Doutriaux and Katie Lewis
Lawrence Livermore National Laboratory
{doutriaux1,lewis66}@llnl.gov

James Ahrens, Benjamin Sims, Christine Sweeney, Divya Banesh, Quincy Wofford
Los Alamos National Laboratory
{ahrens,bsims,cahrens,dbanesh,qwofford}@lanl.gov

SAND2023-08642

EXECUTIVE SUMMARY

A robust data-management infrastructure is a key enabler for National Security Enterprise (NSE) capabilities in artificial intelligence and machine learning. This document describes efforts from a team of researchers at Sandia National Laboratories, Los Alamos National Laboratory, and Livermore National Laboratory to complete ASC Level II milestone #8854 "Assessment of Data-Management Infrastructure Needs for Production use of Advanced Machine learning and Artificial Intelligence."

Milestone Description

This milestone will address the need for infrastructure to support Advanced Machine Learning (AML) workflows and data curation for integration into production simulation work. The ASC program has made initial investments in infrastructure for AML workflows and data curation. This milestone will identify gaps and propose improvements to address these gaps. These findings and recommendations will be documented and shared with the broader ASC community.

Completion Criteria

1. Develop documentation of the tri-lab requirements for a data-management software infrastructure to facilitate a large ASC AML program.
2. Each lab provides an example application/workflow to establish how the data-management software infrastructure would be used. Identify gaps in currently available software tools.
3. Develop a preliminary execution plan for future tri-lab data-management infrastructure investments
    a) Identify common requirements and collaboration areas across the tri-lab community.
    b) Stretch goal: Identify additional requirements to stand up a tri-lab data repository for sharing ASC (possibly OUO/ECI) data sets

Summary of Work Done

We believe the contents contained in this report provide sufficient evidence to satisfy the completion criteria for milestone #8854.

Completion Criteria 1: Chapter 4 provides a list of tri-lab requirements derived from gaps in our current infrastructure and anticipated needs from NNSA strategy documents.

Completion Criteria 2: Chapter 2 provides a rigorous assessment of nine representative AML workflows (three from each laboratory). Chapter 3 identifies limitations/gaps in existing data infrastructure across the different layers of the *Data Technology Stack* (defined in Chapter 1 Section 1.1.1).

Completion Criteria 3: Chapter 4 provides detailed **requirements** for an anticipated complex wide *Data Ecosystem* to enable AML workflows that span the NSE. Chapter 5 provides an

execution plan that includes a mix of institutional and program collaborations and R&D in key areas – addressing criteria 3-a. Section 5.3 proposes (and describes requirements for) development of a repository intended to provide federated access to curated, potentially sensitive or classified, datasets from across the NSE complex. This "ASC Data Explorer" prototype could provide a significant step toward a comprehensive Data Ecosystem. We believe this description satisfies the stretch goal identified in criteria 3-b.

Path Forward

Chapter 5 outlines a high-level plan for building a production ASC Data Infrastrucure that spans the NSE complex. Successful execution will require more than just R&D – it will depend on committed institutional and program partnerships, cultural changes within our project teams that encourage and incentivize data-management plans, and it will require R&D and co-design to address a large set of existing and emerging technical challenges.

ACKNOWLEDGEMENTS

This report included contributions from a number of excellent staff and management from Los Alamos National Laboratory, Sandia National Laboratories, and Los Alamos National Laboratory.

We thank LANL application participants Oleg Korobkin, Soumi De, Marc Klasky, Dru Renner, Gabe Rockefeller, Carola Ellinger, and Raffi Yessayan. We also thank Scott Doebling, Mike Ham, Vanessa Feagin, and Julie Maze of the LANL Weapons Research Services (WRS) group for their input on institutional data management concepts and priorities. We also thank the current DSI team led by Jim Ahrens: Divya Banesh, Hugh Greenberg, Jesus Pulido, Ben Sims, Christine Sweeney, Terry Turton, and Quincy Wofford. The LANL work was funded by the Advanced Simulation and Computing (ASC) program through the Data Science Infrastructure and Integrated Codes projects at LANL. This work was performed under U.S. Government contract 89233218CNA000001 for Los Alamos National Laboratory (LANL), which is operated by Triad National Security, LLC for the U.S. Department of Energy/National Nuclear Security Administration.

Sandia information was provided by Erin Acquesta, Dan Bolintineanu, Ashley Fate, Foss Friedman-Hill, Anthony Garland, Craig Hamel, Brenna Hautzenroeder, Kyle Johnson, Sharlotte Kramer, Reed Milewicz, Zachary Morrow, Kyle Neal, Ben Schwaller,and Nick Winovich. Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA000352

LLNL's ICECap use case information was provided by Luc Peterson, David Strozzi, Andrew Gilette. LLNL's Climate use case information was provided by Sasha Ames, Karl Taylor, Paul Durack and David Bader. LLNL's edge computing use case information was provided by Derek Mariscal, Brian Spears, Timo Bremer and Tammy Ma. LLNL's work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC.

We would also like to thank members of the milestone review committee for taking time to review our content and provide feedback. They include:

- Jaideep Ray (Chair), Sandia National Laboratories,
- Mike Wickett, Lawrence Livermore National Laboratory,
- Gabe Rockefeller, Los Alamos National Laboratory, and
- Lloyd Arrowood, Y-12 National Security Complex.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION AND BACKGROUND

The rapid evolution in AI technology and the massive potential of AI methods for the DOE and NNSA inspired a number of recent strategic activities that emphasize data-infrastructure investments as a critical need. For example, the ASC Artificial Intelligence for Nuclear Deterrence (AI4ND) Strategy [2] mentions *Scalable and Performant Data Infrastructure* as one of four critical investment areas to enable the effective use of AI for the nuclear security enterprise (NSE). Similarly, the AI for Science, Energy, and Security workshops report [3] identifies the need for a DOE-complex wide *Data Ecosystem* as critical to "fully exploit the potential of AI and drive advances in strategic areas of research." Other national strategy documents like the NNSA computing and simulation strategy documents [4, 5] the Office of Science and Technology Policy's AI Strategic Plan [6], and the recent National Academies report on post-exascale computing [7] all mention data infrastructure as a key requirement for deployment of next-generation AI capabilities.

In an effort to guide investments and coordinate tri-lab R&D in data-infrastructure, a team from Sandia National Laboratories (SNL), Los Alamos National Laboratory (LANL), and Lawrence Livermore National Laboratory (LLNL) proposed an FY23 ASC Level II milestone titled, "Assessment of Data-Management Infrastructure Needs for Production Use of Advanced Machine Learning and Artificial Intelligence" (see Figure 1-1). This report is the *record of completion* for that milestone and includes the following:

- In later sections of this chapter, we define concepts, frameworks, and terminology, based on industry and community standards, to set a baseline for a discussion about data-infrastructure needs. This includes an introduction to the data-technology stack model and a review of data management and governance concepts.

- In Chapter 2, Example Applications and Workflows, we identify gaps in the current data-infrastructure by assessing requirements of representative AI/ML workflows from each of the three NNSA laboratories. This data was collected through structured surveys with application teams. Table 1-1 provides a summary of the examples discussed in this report.

- Chapter 4, Data-Infrastructure requirements to enable production use of AML, provides a discussion of required capabilities for a data-infrastructure designed to support AI/ML workflows. These requirements derive from the gaps identified in the application examples from Chapter 2.

- Chapter 5, Execution Plan, identifies anticipated data-infrastructure needs based on capability needs described in recent DOE strategy documents, and proposes a plan to address current and anticipated needs through R&D investments

Milestone (ID#8554): Tri-lab AML Milestone: Assessment of Data-Management Infrastructure Needs for Production use of Advanced Machine learning and Artificial Intelligence.		
Level: 2	**Fiscal Year**: FY23	**DOE Area/Campaign**: ASC
Completion Date: 9/30/23		
ASC nWBS Subprogram: CSSE		
Participating Sites: SNL, LANL, LLNL		
Participating Programs/Campaigns: ASC		
Description: This milestone will address the need for infrastructure to support Advanced Machine Learning (AML) workflows and data curation for integration into production simulation work. The ASC program has made initial investments in infrastructure for AML workflows and data curation. This milestone will identify gaps and propose improvements to address these gaps. These findings and recommendations will be documented and shared with the broader ASC community.		
Completion Criteria: 1) Develop documentation of the tri-lab requirements for a data-management software infrastructure to facilitate a large ASC AML program. 2) Each lab provides an example application/workflow to establish how the data-management software infrastructure would be used. Identify gaps in currently available software tools. 3) Develop a preliminary execution plan for future tri-lab data-management infrastructure investments a. Identify common requirements and collaboration areas across the tri-lab community. b. Stretch goal: Identify additional requirements to stand up a tri-lab data repository for sharing ASC (possibly OUO/ECI) data sets.		
Customer: ASC		
Milestone Certification Method: A milestone review is conducted, and its results are documented. A set of annotated slides is prepared as a record of milestone completion.		
Supporting Resources: CSSE and FOUS Staff		

Figure 1-1. The Official ASC Milestone #8554 description as recorded in the FY23 ASC Implementation Plan [1].

Table 1-1. Summary of application-workflow examples used to identify requirements and gaps in our current data-management infrastructure.

LANL Examples	
Radiograph	The Radiograph AI/ML example describes a workflow that uses AI/ML training to predict radiograph simulation parameters for shock physics.
Ensembles	The CMF/Ensembles example describes the Common Modeling Framework and how it's used to develop and run ensembles of simulations.
Bueno	The Performance/Bueno example describes workflows using code performance and testing data from LANL's Lagrangian Applications Project.
LLNL Examples	
ICECap	The Inertial Confinement on El Capitan workflow uses AI/ML on the El Capitan HPC system to explore a large design space for Inertial Confinement Fusion (ICF) experiments.
Climate	The Climate Data example from LLNL describes data-management challenges for representative climate-simulation workflows within the Program for Climate Model Diagnostics and Intercomparison (PCMDI).
Edge	The Edge Computing and Real-Time Decision making example describes representative workflows where computational resources interact with experimental devices in real time.
SNL Examples	
PIML	The Physics-Informed Machine Learning Material Models for Solid Mechanics example describes workflows that combine experimental and simulation data to support certification of additively-manufactured parts for the weapons stockpile.
HPC Monitoring	The HPC Resouce Management and Monitoring example describes workflows used to collect and analyze HPC center performance
Wildfire DT	The Wildfire Digital Twin example describes workflows and data-management challenges associated with developing a ML-derived digital twin of wildfires.

1.1. Data Infrastructure

Data infrastructure at a technological level comprises the computing and storage resources required to enable data management activities. This infrastructure must be closely supported by a parallel organizational infrastructure of data governance and management, high performance computing and data management support organizations (and associated subject matter experts), and an organizational culture that encourages responsible and secure sharing of data. A well-designed, well-provisioned, and well-staffed data infrastructure is critical to enabling the flow of data across workflow boundaries in a digital enterprise (to creating federated workflows), especially when the boundaries are inter-institutional.

Relationship to workflows

Numerous workflow management systems exist within each laboratory (though few of these also provide data artifact management), and many data-intensive projects in the laboratories use no formal workflow management system. Workflow management systems are built on top of one or more data infrastructures, but they are distinctly different and are massive undertakings in their own right. Effective workflow tools are highly dependent on the implementation of domain-specific user experiences.

1.1.1. The Data Technology Stack

Data infrastructure may be envisioned as a layered system, or stack, of technologies, with high-level analysis and visualization tools at the top and hardware at the bottom, with several layers of abstraction in between. We propose the following breakdown of layers as a general framework for data-intensive science infrastructure. While many existing tools may fit into one of these layers, newer and more integrated data science infrastructure systems may be designed deliberately to span several technological layers.

Analysis and visualization tools comprise a variety of analysis and visualization tools used for advanced AI/ML workflows as well as other science and engineering applications. These are the tools that are typically used by domain scientists to analyze data to produce meaningful results, and to communicate those results within the research community. They encompass a range of tools from basic data analysis and scientific software libraries, including those used for AI/ML, to end user-facing tools like Paraview, Cinema, etc. Most scientific workflow management tools also operate at this level, although they may have features applicable to other layers as well.

Data interfaces provide an organization for data such as a description of its structure, type and meaning. We include the programming languages and software libraries that provide the underlying framework for data descriptions that enable data analysis, visualization, and AI/ML workflows. We include tools in our definition that support data/metadata access and manipulation, restructuring, and cleaning of data. Examples include programming languages generically (e.g., Python, C++, R, FORTRAN); software libraries that enable basic data access and manipulation prior to scientific analysis, such as PANDAS, SQLAlchemy, and RSQLite; and data formats such as SQL and HDF5.

Data services are software and hardware used for maintaining, organizing and moving data. Examples include database management systems, data warehouse services, data transfer services, and metadata support and indexing tools. These are the front-line tools for enabling FAIR data access. While analysis, visualization and ML tools often enable specific workflows, these data service resources serve to manage data in a general way, potentially making them available to many users for a variety of preprocessing and analysis workflows. Data services are typically standardized and provided at an institutional level.

Data architecture includes common tools and standards for enabling interoperability of data and data analysis tools, potentially both within and across institutions. These are often associated with institutional data management and governance standards and provide the fundamental capabilities necessary for FAIR data access and other institutional requirements such as security. They include API libraries (i.e., institutional repositories of supported APIs), metadata frameworks, data virtualization services, and security frameworks (e.g., POSIX security). The term architecture reflects the need to design these systems as an environment for support broad institutional needs rather than those of a specific research group.

Storage, networking, and computing systems encompass the hardware infrastructure and fundamental protocols and formats that enable data storage, transfer, and use in computing environments. Protocols and formats cover types of storage (file system, object storage, etc.) and data transfer protocols, which are in turn closely associated with the actual hardware for storage and networking. In an HPC context, these systems are also closely associated with and tuned to requirements for I/O, memory, and other aspects of machine architecture and managed at an institutional level. In other cases, storage, networking, and computing systems may be built to less specialized requirements or be more locally controlled by a research team or group.

1.1.2. Organizations, Culture, and Workforce

Organizations: At an organizational level, ensuring convergence between researchers and institutions over control of data point to the need for strong, negotiated data governance principles that assert institutional control over data while respecting researchers' autonomy and ability to ensure data quality and security. It may be helpful to look to larger research, systems engineering, etc. projects where institutional control over data is stronger for examples of how data can be successfully shared in a context of institutional governance. Data management and governance are discussed in more detail below. Another crucial organizational function is providing training and support for implementation of new tools and infrastructure, which may be important to addressing some of the cultural issues described below.

Culture: Successfully implementing a shared infrastructure where data is made widely available across projects and organizations (within security constraints) will require significant cultural changes in the national laboratories, particularly around control and "ownership" of data. The current tacit assumption among many researchers and managers at the national laboratories is that PIs and research teams in some sense "own" the data they generate, i.e., have control over when and how this data may be accessed, and by whom. There are good reasons why researchers should have some control over data, particularly in an environment where lack of good metadata or other context around data sets could lead to misinterpretation of their content. Scientific norms also provide that researchers retain some degree of control over data to support independence of scholarship and priority of discovery. In addition to these concerns, research and engineering teams are often simply reluctant to adopt major changes to workflows that are already working well for their specific purposes.

The assumption of local “ownership” of data contradicts the actual terms of employment at the national laboratories, which typically dictate that all research outputs belong to the institution. The current lack of shared data infrastructure has prevented this contradiction from being a source of conflict, because from a practical point of view there has typically been no way for researchers to gain access to another research team’s data without direct communication with that team. A future data technology stack that removes these barriers without appropriate need-to-know and embargo periods could generate conflict between researchers and institutions over data sharing in the absence of broader negotiations to articulate new norms for data sharing and control that are acceptable to all parties.

Another common cultural obstacle to sharing data is a lack of “trust” in data generated by someone else. Again, this can be a reasonable concern, especially when the data lack good metadata or the provenance of the data is unclear. Nonetheless, the need for copious amounts of diverse data for AI-driven solutions necessitates the ability to leverage data from other research groups. Improvements to how we create, annotate, and protect data from corruption will be necessary to build community trust to use data generated by others.

Both of these concerns – ownership of data and lack of trust in data generated by others – are similar to experience that we have already faced with the increase in library use and open-source software. It is important to use that experience to influence culture related to effective data sharing.

Workforce: The widespread adoption of new data sharing norms, tools, and hardware will require substantial shifts in the workforce required to support these elements. In particular, it will create a need for data scientists, as well as librarians, archivists, and data management professionals with strong computing skills and training in the unique needs of data-intensive scientific computing and advanced AI/ML researchers. Among domain scientists themselves, it may put a premium on those who have experience with managing, staging, processing, and analyzing large data sets, including in the context of AI/ML workflows. High performance computing support organizations may also need additional data specialists to support increasingly data-intensive HPC workflows.

1.2. Data Management and Data Governance

Data management covers the tools, practices, and policies that are needed to support “the development, execution, and supervision of plans, policies, programs, and practices that deliver, control, protect, and enhance the value of data and information assets throughout their lifecycles." [8] As such, it is closely related to the activities supported by data infrastructure, such as capturing, organizing, storing, and preserving data. Data management and data infrastructure are interdependent systems that together create a data ecosystem that meets the needs of institutional and individual stakeholders. Any organization that handles data engages in data management and governance, whether intentionally with an eye toward maximizing the value of data, or less formally through the individual activities of its members.

Data governance is the element of data management that directly concerns the institutional "exercise of authority" to set standards and policies for data management within an organization [8, 9, 10]. This primarily relates to the high-level policies, procedures, and leadership actions that set institutional expectations about how members will manage data and who has the authority to make decisions about how data will be maintained, stored, preserved, and documented. This need not imply a rigid top-down approach; successful data governance draws on broad input from across institutions and serves the needs of a wide range of stakeholders, from leadership down to the level of data producers and consumers. In particular, in settings like the national laboratories where data is integral to cutting-edge, computationally intensive science and engineering efforts, data management and governance must take careful consideration of the day-to-day needs of data producers and consumers so it can enhance and add value to their work rather than hindering productivity. To accomplish this, it is important to back up data management and governance with user-friendly, performant data infrastructure and technical support that make it easy and desirable for data producers and consumers to follow institutional policies.

At the national laboratories, data management and governance functions are currently often scattered across many organizational units at different levels of the organization, and often come down to decisions made by individual project leaders or PIs about how to manage data used or produced by their own projects. This status quo will not be sufficient to govern new data infrastructures that cross organizational or institutional boundaries to support advanced AI/ML workflows. There will be an increasing need to replace ad-hoc data management approaches with more formal, institutionally negotiated approaches that meet the needs of all stakeholders. These concepts are discussed further below.

1.2.1. The FAIR Principles

The FAIR principles (Findable, Accessible, Interoperable, and Reusable) are a set of widely accepted guidelines for the governance and management of scientific and engineering data [11]. We believe these guidelines provide essential grounding principles for design of data infrastructure to support advanced AI/ML workflows. Although they do not address every requirement in this space, they provide an essential foundation for building tools and processes that are more specialized toward advanced AI/ML workflow needs, and are not widely supported by current tri-lab infrastructure. The FAIR principles are specifically intended to support machine discovery and access to data "to improve knowledge discovery through assisting both humans, and their computational agents, in the discovery of, access to, and integration and analysis of, task-appropriate scientific data and other scholarly digital objects." [11]. In addition to supporting research, these principles can help organizations preserve and manage data for the longer term and inform institutional or cross-institutional data governance decisions.

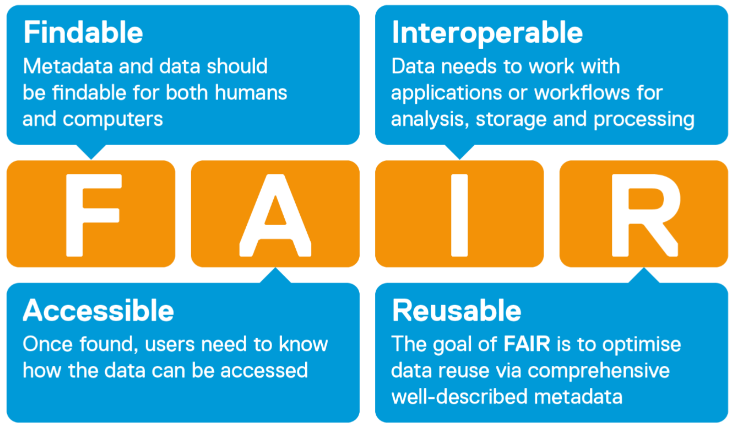


Figure 1-2. FAIR principles (Findable, Accessible, Interoperable, and Reusable) [11] are widely adopted in the open science community. Image from `https://scibite.com/solutions/enterprise-fair-data-mdm/`.

Each of the elements of FAIR (illustrated in Figure 1-2) are highly relevant to advanced AI/ML workflows. Making data findable requires that data be associated with persistent identifiers and rich metadata describing the data, and that this data and metadata be indexed in some way. This is particularly important for automated AI/ML workflows where a computational agent may need to make decisions about relevant data without intervention from human subject matter experts. Making data accessible requires standardized or interoperable protocols for retrieving data while supporting appropriate authentication and authorization mechanisms. Again, this is crucial for enabling automated access to data resources for advanced AI/ML workflows without compromising security. Data also needs to be interoperable, in the sense that mechanisms exist for readily integrating it with tools and other data. This may require standard vocabularies and languages for knowledge representation so that, for example, particle velocity in one experiment may be mapped or compared to particle velocity in another even if the original data labels were not identical. This interoperability is essential to enable AI/ML tools to automatically build connections and traverse diverse data sets. Finally, data is reusable if it meets requirements for future use in relevant science and engineering fields, which again requires rich metadata, as well as clear rules and licensing for sharing data and detailed provenance information so data can be reliably traced to its origins. This is essential to enable long-term reuse and mining of data in AI/ML workflows, and is an important part of making AI/ML workflows explainable with reference to data origins.

The FAIR principles apply to data, metadata, and elements of the data infrastructure stack described above. For example, the guidance that metadata be "registered or indexed in a searchable resource" has direct relevance to the data architecture and interoperability and data services layers of the data technology stack, while also having implications for data governance (e.g. institutional requirements for investigators to register metadata) and data management (e.g. how to set up workflows that create and use the metadata indexed in these resources).

Making data FAIR is not a trivial undertaking. It is likely to require significant investments in data infrastructure and knowledge-sharing frameworks at a scale that does not currently exist in the national laboratory complex. While emerging AI technologies may play a role in linking data, generating metadata, and other interventions that support FAIR principles, and may reduce the level of effort required for some of these interventions, these AI capabilities are still emerging and may not be production-ready for some time; in addition, they will themselves depend on data being findable, accessible, interoperable, and reusable at some level in order to begin bootstrapping their own foundational data models. Significant investments in data management, data governance, and data infrastructure are required whether it is human or AI data librarians traversing and connecting data sets to support advanced AI/ML workflows.

At a more detailed level, FAIR principles lay out a set of requirements that are directly applicable to advanced AI/ML workflows, and are applicable whether humans or AI systems are discovering and organizing the data required for these workflows. For example, regardless of how it is achieved, advanced AI/ML workflows will require that data be described with rich metadata (principle F2) and assigned a globally unique and persistent identifier (principle F1).

1.2.2. Metadata

Metadata is a key aspect of building a successful data infrastructure and data management framework. The most basic definition of metadata is "data that define and describe other data." [12]. In this sense, metadata could cover a wide range of data that relates to other data. It is often analogous to the data fields or column headings in a database. More specifically, in the context of knowledge management and preservation, it usually refers to "structured information that describes, explains, locates, or otherwise makes it easier to retrieve, use, or manage an information resource" (NISO 2004) [13]. If standardized and structured via defined schema, metadata is machine readable. Well-defined metadata is crucial to discovering, understanding, and making use of data sources, particularly in large, cross-cutting data repositories and for automated AI/ML workflows where contacting the data originator may not be practical.

A key element of the definition above is the term "structured": although metadata can refer to almost any data relating to other data, in practice what we call metadata is usually organized into schemas, registries, or repositories that define a common set of shared metadata elements and their relationships to one another. There are many standards that support these metadata structures, such as the ISO/IEC 11179 Metadata Registry Standard [12]. These might be organized at an institutional level or in the context of a particular data generation effort or research project where specialized metadata elements are needed to meet more localized needs. The level at which metadata is standardized is relevant to the ability to achieve data interoperability. For example, an institutional data repository might standardize widely used metadata elements like creator, title, date, format, etc., elements of the commonly used Dublin Core metadata standard (ISO 15836-1:2017). A more specialized metadata schema might include elements like "run number," "production script," or "luminosity block number," which were defined for data from the ATLAS experiment at CERN and are useful primarily to members of that collaboration [14]. Due to the specialization and technical specificity inherent to data-intensive research environments, this kind of specialized metadata plays a much larger role that it does in general enterprise data management.

Type of Metadata	Definition	Examples relevant to infrastructure
Descriptive	Enables discovery, identification and selection	Title, author, and keywords
Administrative	Facilitates the management of resources	Location, size, and access permissions
System-related (technical)	Enables decoding and rendering of data	Data format and precision, documentation needed for reading data
Discipline-specific	Facilitates interpretation and reuse of data by domain specialists	Various domain-specific data descriptors
Specific to AI/ML/data-intensive science and engineering	Supports reproducibility and interpretation of data by a range of skilled users	Input deck parameters and data, code and compiler information, machine learning parameters, uncertainties

Table 1-2. Types of Metadata

Metadata is, in part, a semantic resource that can be used to define key concepts in a particular area of data management. It is similar to – and can be connected to – other semantic resources an organization may choose to maintain, such as dictionaries, thesauri, taxonomies, knowledge graphs, or ontologies. These can help further define a shared information space and its common elements. Many enterprise data and knowledge management tools provide for a central organizational semantic store that provides authoritative terminology sources that a variety of data and knowledge management tools can draw on to ensure interoperability.

There are many different ways of categorizing metadata into different types [15]. Some of the most important are summarized in Table 1-2. These range from administrative and system-related metadata that are mainly useful to data specialists, to general descriptive metadata, to discipline-specific metadata that provides fine-grained information useful primarily to domain specialists. Many disciplines and even individual research projects have their own data standards that support these specialized uses. However, to support cross-institutional data-intensive science and AI/ML workflows, there may be a need to define a set of common metadata elements to assist with reproducibility and interpretation of data by researchers in these fields and by domain scientists who use these methods. It does not appear that such a standard currently exists.

1.2.3. Security and Access Control

The FAIR principles and many metadata standards are oriented toward open dissemination of data, and specifically with maximizing the availability of data. In a national security setting, however, security requirements place complex constraints on how data can be made available and to whom. In this context, the access element of FAIR must emphasize mechanisms for reliably controlling access rather than maximizing access in all circumstances. This entails maximizing availability of data but only within a constrained scope of access governed by information sensitivity, access permission, and need-to-know policies. This may be further complicated when AI/ML systems build models drawing on datasets that may have different access control requirements, or where compilation of data elements has the potential to increase data sensitivity beyond the sensitivity of the original source datasets. The most straightforward solutions in this area will leverage existing access control schemes such as standard POSIX access controls.

2. EXAMPLE APPLICATIONS AND WORKFLOWS

This chapter provides an assessment of the existing data infrastructure at the NNSA laboratories by evaluating how a selection of applications and workflows manage their data through various phases of the *research data lifecycle* model [16]. The research data management lifecycle (see Figure 2-1) describes the various stages of research data for a typical research or engineering project. While there are similar data lifecycle models that describe institutional data management processes, a research-focused model is the most representative of projects at the NNSA and is the basis for the DOE policy for Digital Research Data Management [17] and motivated by numerous U.S. Federal Advisory Committee Reports [18, 19, 20, 21, 22, 23].

Figure 2-1. Research data typically evolves through a series of phases described by the research data management lifecycle [16].

The elements of the research lifecycle include:

- **Plan and design**: Determine research goals and methods; identify required data infrastructure and create data management plan
- **Access or acquire**: Make data accessible for future use via metadata and user-friendly infrastructure; set access control policies
- **Process and analyze**: Process data; analyze data in light of research goals; derive conclusions
- **Archive**: Select data for long-term preservation
- **Share and publish**: Publish results and/or make appropriate data publicly accessible
- **Store and transfer**: Store and transfer data as needed. Each stage of the lifecycle has different requirements in terms of amount of storage, transfer speed, frequency of access, latency, etc.

2.1. Los Alamos National Laboratory

2.1.1. Overview of LANL AI/ML Use Cases

LANL has a number of AI/ML use cases that are rapidly evolving. Several of them involve using AI/ML to help parameterize simulation models or models like strength models that are used as input to simulation models. The goals for better model parameterization include, but are not limited to: 1) improve the quality of simulation results by overcoming errors or limitations of the simulations, 2) expand the applicability of the simulations to different materials or conditions, 3) improve the ability of the simulations to produce results in line with experimental results, 4) scale simulation fidelity properly or at least more optimally, and 5) reduce time to solution for expensive simulations.

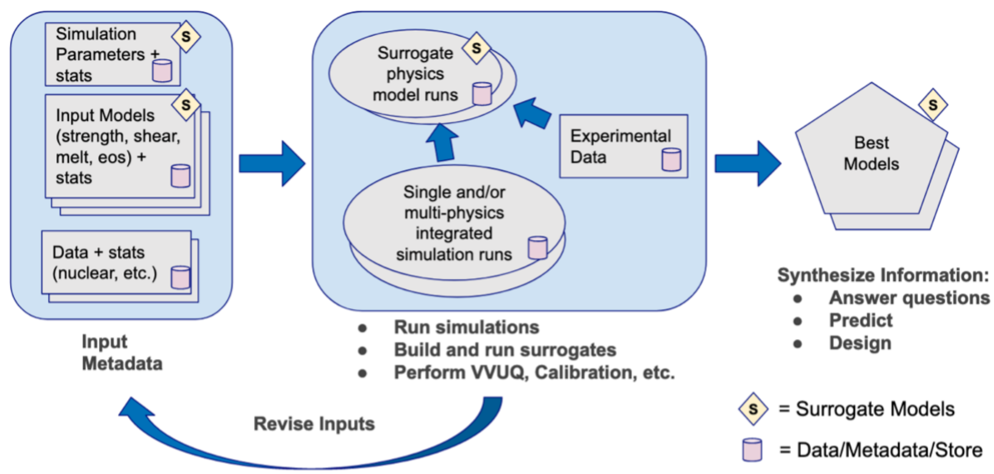


Figure 2-2. An example workflow used to build ML-generated surrogate models. Image by C. Sweeney.

Figure 2-2 shows a generalized workflow where AI/ML is used to build surrogate models that can be run on their own, provide better input parameters to simulation runs, help compare to experiment, or used to choose the best simulations to run. This, in turn, results in a better set of models (surrogate and/or full simulations) from which information can be gleaned via answering questions, making predictions or guiding design. Along these workflows, various data and metadata fields are highlighted in the diagram as well as resulting AI/ML models that are also considered to be data products. A few LANL AI/ML use cases are described next.

A use case for AI/ML to improve quality of results is the EUCLID project [24], where statistics on nuclear data are used to reduce errors that limit simulations that use nuclear data. For the Impala project [25, 26], statistical emulators (AI-based surrogate models) are used to create probability distributions for parameters to strength models that help provide better strength models to FLAG simulations for different conditions and/or materials. An example of AI/ML used to improve calibration of simulations with experiment are the neural networks trained on shell implosion simulations to help predict densities of radiograph data collected at the Dual-Axis

Radiographic Hydrodynamic Test facility (DARHT)[1] . AI/ML to help with simulation fidelity tuning is planned for high-explosive models. For quantum chemistry, AI/ML is used to reduce the computational cost of quantum mechanical simulations via active learning [27]. Beyond simulation use cases, AI/ML-based analytics are being used to assess the quality of manufactured parts as part of the CACTI CMM Analytics Toolkit.

2.1.2. Radiograph Simulation Parameter Prediction

2.1.2.1. Plan and Design

ASC has a number of simulation and data analytics efforts in the domain of radiograph data. One of the efforts uses an ML-based technique to reconstruct density fields in dynamic systems based on a temporal sequence of radiographic images obtained by robust hydrodynamic simulations. They use CTH [28] hydrodynamics simulations of 1D and 2D shell implosions to create datasets to train neural networks (NNs) that can predict input parameters to CTH given shock and edge features – the inverse problem. The idea is to input the shock and edge of an experimental radiograph into these NNs to infer CTH parameters, then run the CTH simulation to generate detailed density fields. The workflow is depicted in Figure 2-3. The Data Science Infrastructure (DSI) project is using this machine learning workflow as a use case for harvesting, storing and querying metadata for these large simulation datasets and accompanying analysis and configuration artifacts [29]. Note that this is just one AI/ML workflow out of other previous, current, or planned AI/ML workflows for radiograph data for this and related projects.

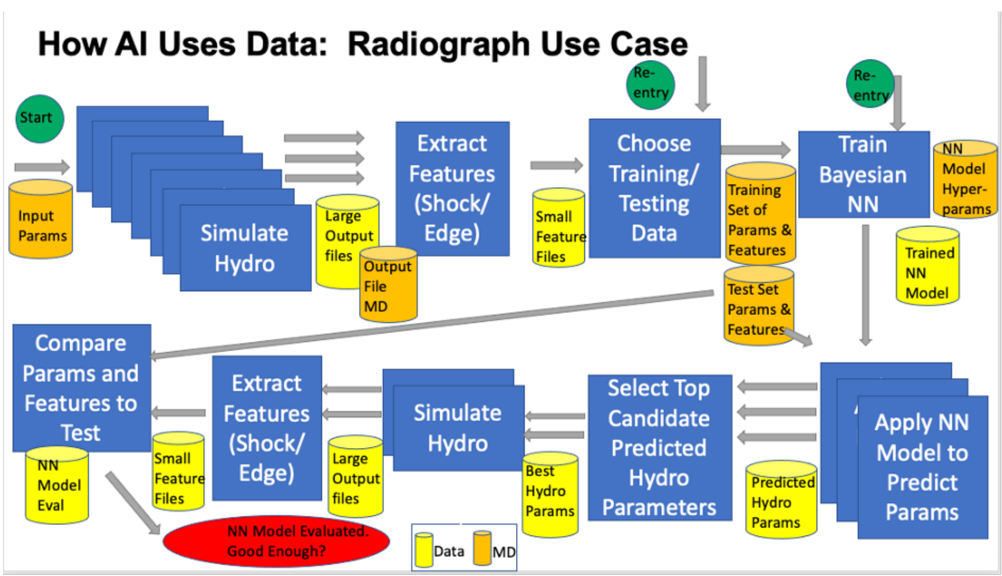


Figure 2-3. Workflow diagram of AI/ML training to predict radiograph simulation parameters that will match desired shock and edge features. Image by C. Sweeney.

[1]Dual-Axis Radiographic Hydrodynamic Test facility: https://science-innovation.lanl.gov/science-facilities/darht/

2.1.2.2. Acquire

The simulation data are produced by CTH runs and made available for analysis. These datasets used for training the neural networks can run on the order of 160K and 10K files for an ensemble of 1D and 2D simulations respectively [30]. Example simulation data is shown in Figure 2-4.

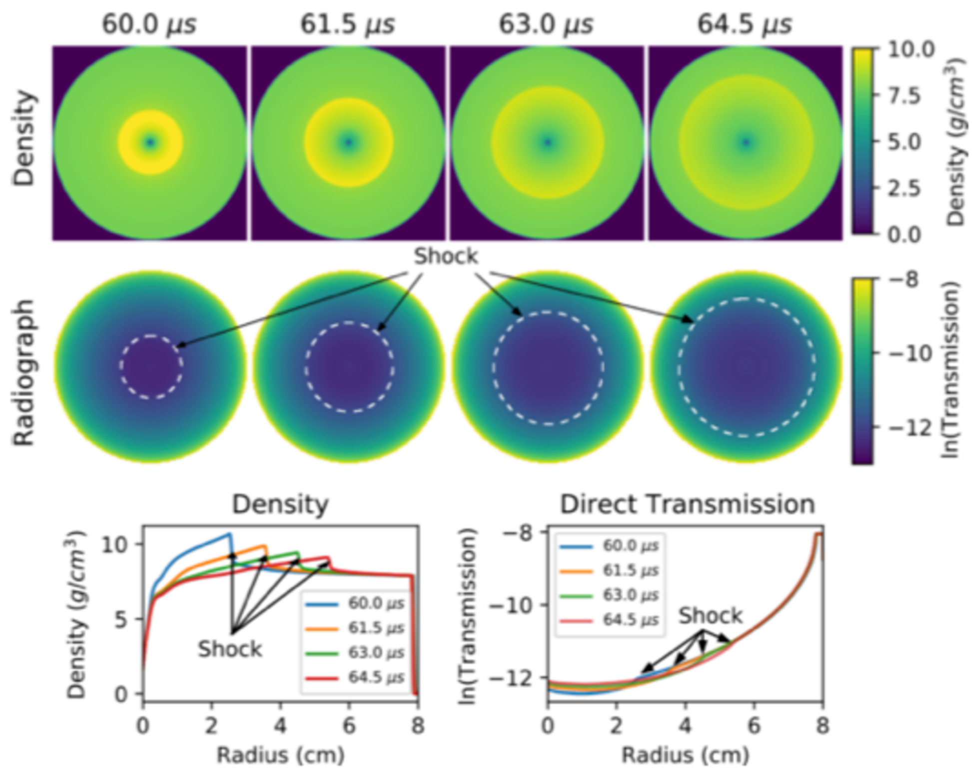


Figure 2-4. Simulated radiograph data at different time steps showing shock and edge features. Image from [30].

.

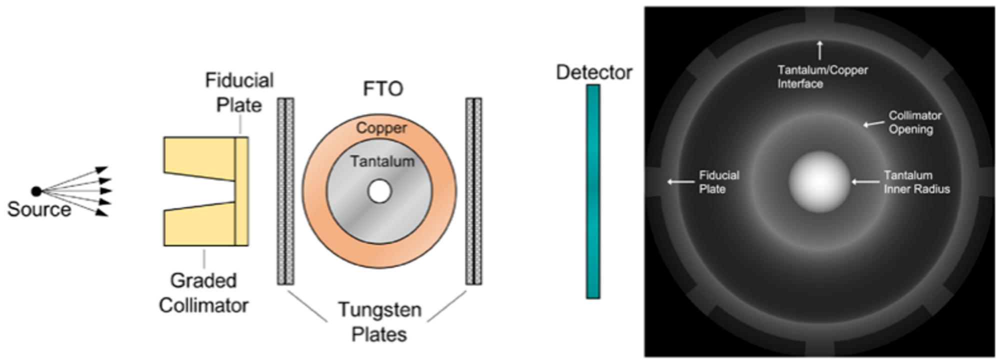
The experimental data are acquired at a facility like DARHT as shown in Figure 2-5.

2.1.2.3. Process and Analyze

Raw data from the runs is organized into directories, 133 directories for 1D, 155 for 2D. Examples are shown in Figure 2-6. Directories are named to coincide with parameter values. Parameter values are given labels, so are not just numbers. The label maps to a value and the number maps to percentage of that value, which seems to be how the parameter sweeps are performed.

Usually they also build a summary space-separated file (features.dat) of all the CTH parameters for the runs and the resulting shock and edge feature values for each run. They use Python tools to build the features.dat file. An example of the features.dat is shown in Figure 2-7.

Sometimes they also generate images of the output of each run to see the shock and edge visually as shown in Figure 2-8.



(a) (b)

Figure 2-5. (a) Schematic of the DARHT radiographic scene. (b) French Test Object simulated radiograph. Image from [31].

.

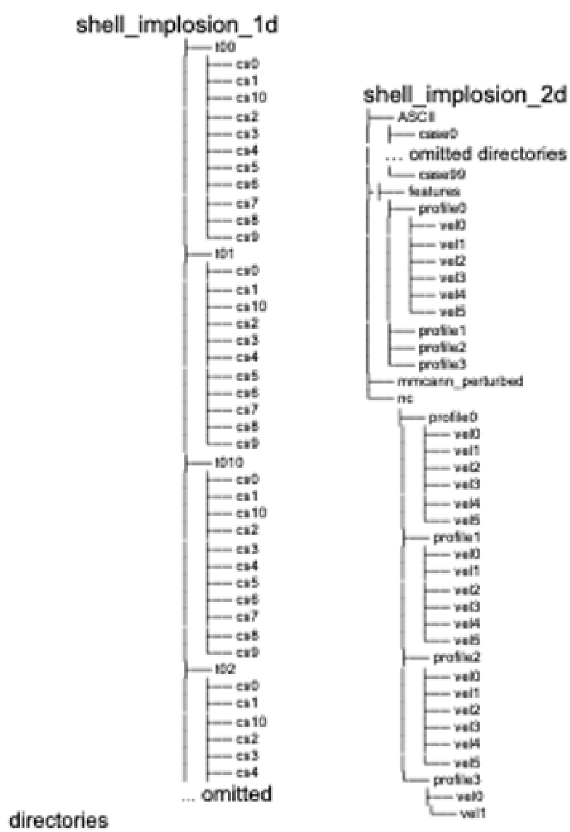


Figure 2-6. Example directory hierarchies manually organized to store raw CTH simulation data.

```
# Fits to the temporal evolution of the 0th harmonic (average radius) for the shock and edge for 2D shell implosion simulations
#
# WARNING: there are supposed to be 4*6*5*3*4*3*4 = 17280 simulations in this database, but there are actually only 10293 of them.
# This file is sorted in numerical order in the first five columns, but the missing values are simply omitted.
#
# Fields:
# 1:profile [0..3]                profile type: see simlation-analysis-tools/shell_implosion_2d/profiles for exact shapes
# 2:vel [0..5]      initial implosion velocity: {0:95000, 1:94335, 2:94620, 3:94050, 4:95950, 5:95475} cm/s = {0:100%, 1:99.3%, 2:99.6%, 3:99%, 4:101%. 5:100.5%}
#   3:mgrg0 index [0..4]     Mie-Grueneisen G0: {0:1.6,   1:1.7, 2:1.76, 3:1.568, 4:1.472}                  = {0:100%, 1:106.25%, 2:110%, 3:98%, 4:92%}
#     4:s1 index  [0..2]     Mie-Grueneisen S1: {0:1.22, 1:1.464, 2:1.342}                                = {0:100%, 1:120%, 2:110%}
#       5:cs index [0..3]    MG speed of sound: {0:3.39, 1:3.729, 2:3.051, 3:3.55} um/ns                  = {0:100%, 1:110%, 2:90%, 3:104.72%}
#         6:cv index [0..2]  MG parameter  C_V: {0:1.6, 1:1.76, 2:1.44} x 1e10                            = {0:100%, 1:110%, 2:90%}
#           7:ptwg index [0..3]    PTW G param: {0:6.5e11, 1:6.5e12, 2:6.5e10, 3:6.5e9}                   = {0:100%, 1:1000%, 2:10% 3:1%};
# --------------|--------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------
#profile        |8:shock: chi-square/dof for the parabola fit a_s*(t-t0)^2 + b_s*(t-t0) + c_s, t0 = 40us           | 15:edge: chi-square/dof for the parabola fit a_e*(t-t0)^2 + b_e*(t-t0) + c_e, t0 = 40us
# vel s1  cv    |        +--------------+--------------+--------------+--------------+--------------+--------------+         +--------------+--------------+--------------+--------------+--------------+--------------
#   g0  cs  ptwg|         9:a_s[cm/us2] 10:sigma a_s   11:b_s[cm/us]  12:dekta b_s   13:c_s[cm]     14:sigma c_s |          16:a_e[cm/us2] 17:sigma a_e   18:b_e[cm/us]  19:dekta b_e   20:c_e[cm]     21:sigma c_e
# --------------------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------
0 0 0 0 0 0 0  0.08442  -5.6394821e-03  2.0419071e-06  5.7283884e-01  6.1937848e-07  1.9797820e+00  2.5129070e-07  0.00310  6.7230799e-04  3.4927803e-08  -3.1619735e-02  1.0089085e-08  7.6900883e+00  8.3611910e-09
0 0 0 0 0 0 2  1.22538  -1.5804542e-02  2.9638587e-05  5.8502518e-01  8.9903714e-06  2.2296708e+00  3.6475221e-06  0.00446  2.4060403e-04  4.9708255e-08  -2.8976555e-02  1.4481672e-08  7.6609252e+00  1.1992945e-08
0 0 0 0 0 0 3  1.05380  -1.8431537e-02  2.5386523e-05  5.8610185e-01  7.7397141e-06  2.2561412e+00  3.1363361e-06  0.00320  9.1821015e-04  3.5498198e-08  -2.7964609e-02  1.0361863e-08  7.6571365e+00  8.5997282e-09
0 0 0 0 0 1 0  0.08735  -3.4811868e-03  2.1127925e-06  5.7168056e-01  6.4088040e-07  1.9784920e+00  2.6001433e-07  0.00376  4.8263945e-04  4.1894049e-08  -3.1788471e-02  1.2205133e-08  7.6908365e+00  1.0107638e-08
0 0 0 0 0 1 2  1.04325  -1.2741695e-02  2.5247862e-05  5.8447570e-01  7.6461198e-06  2.2288069e+00  3.1089178e-06  0.00407  2.9405079e-04  4.5414195e-08  -2.9503610e-02  1.3207940e-08  7.6603465e+00  1.0957683e-08
0 0 0 0 0 1 3  1.81059   1.4201404e-02  4.3881704e-05  5.6672430e-01  1.3278064e-05  2.2551148e+00  5.4028460e-06  0.00299  3.3829471e-04  3.3382909e-08  -2.7384210e-02  9.7011110e-09  7.6554351e+00  8.0584079e-09
0 0 0 0 0 2 0  0.07366  -4.3078006e-03  1.7816607e-06  5.7400010e-01  5.4043706e-07  1.9881997e+00  2.1926304e-07  0.00344  7.6833056e-04  3.8372813e-08  -3.1556427e-02  1.1179280e-08  7.6893376e+00  9.2580807e-09
```

Figure 2-7. A snippet of the features.dat file showing parameters (columns for profile, vel, s1, cv, g0, cs, tpwg) and shock and edge features (columns 8-14 and 15-21 respectively). Data format description is at the top of the file.

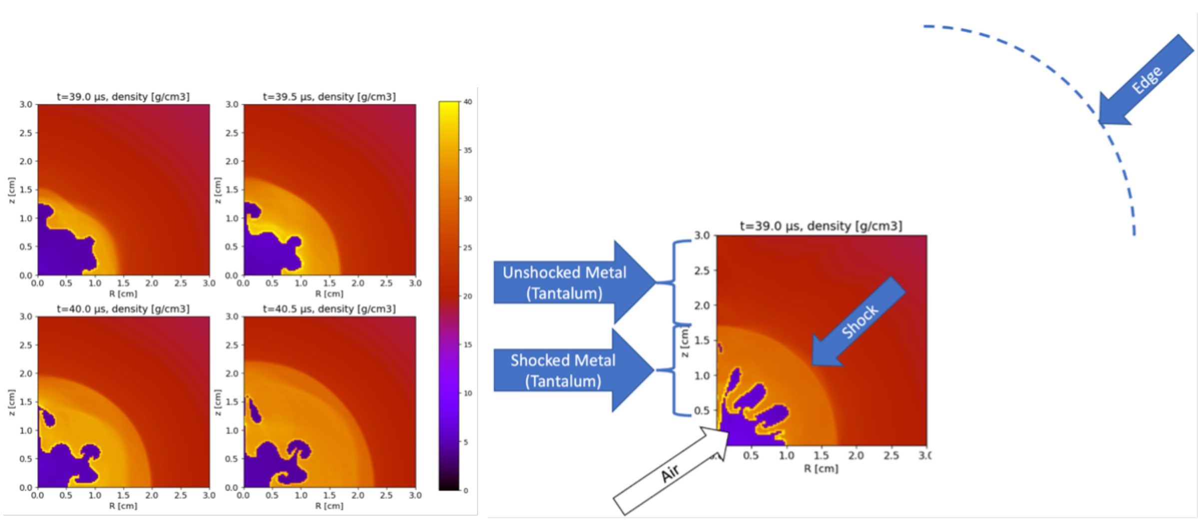


Figure 2-8. Python-generated images (from CTH simulation data) for four time steps (left). Schematic showing the shock and edge features explicitly, where edge is not contained in the images generated by the Python script (right). Image by C. Sweeney.

Training data subsets are produced with different methods, but there is no formal mechanism for tracking subsets.

As part of the DSI project, we produced a tool (see Figure 2-9) that shows the metadata for the ensembles of CTH runs and compares the metadata to a ground truth. This difference is shown so users can manually create subsets of data that can be part of a training dataset. The tool records the analysis and training dataset choices so that the user can go back and further refine a training dataset.

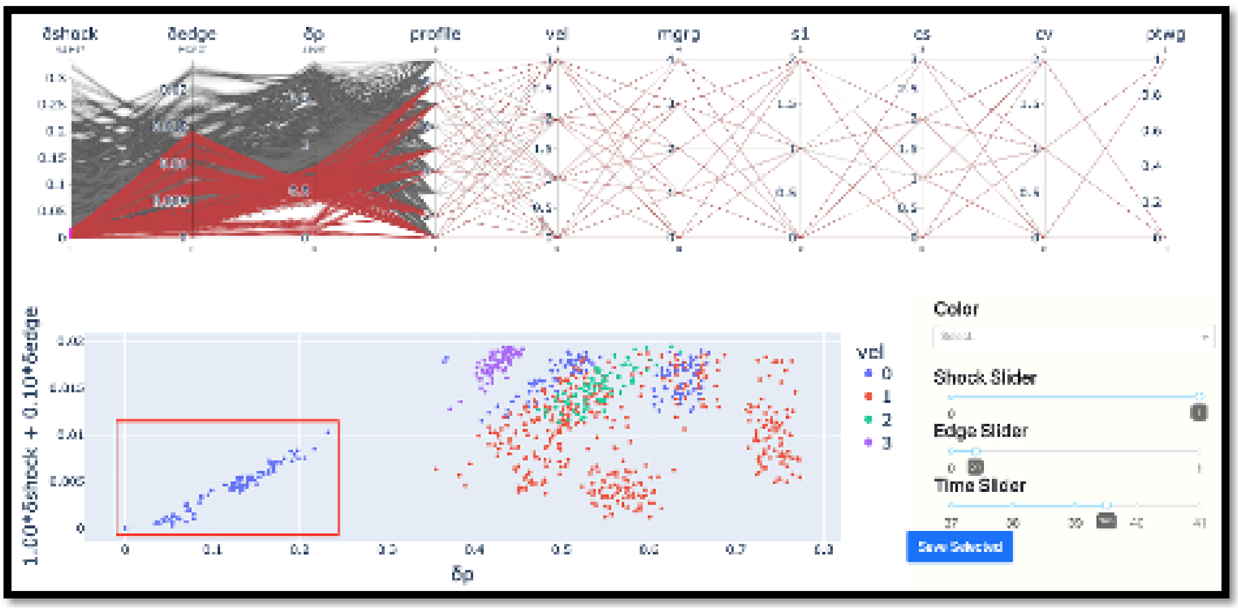


Figure 2-9. Radiograph metadata selection tool main window. Image by O. Korobkin.

2.1.2.4. Archive

They currently have a workflow where they can store two large ensembles of runs. One ensemble is being processed and the other is being generated. It can take two weeks to generate an ensemble. Once the generated ensemble is complete, they usually are finished processing the previous one, so they delete it and keep the analysis. Then they start analyzing the new ensemble and start generating a new ensemble again.

A long-term storage scheme is unknown.

2.1.2.5. Share and Publish

A valid subset of run results are used for analysis. They use group permissions to restrict access to the data.

As mentioned previously, raw data from the runs is organized into directories that are named by CTH simulation parameters, so are easier to navigate than one flat directory, but makes it difficult to share, search and add data, because it is fairly fixed. It is possible to get the handle of a particular data file if you know how to navigate the directories, but there is no direct lookup of the file path given the parameters. Parameters and resulting features are kept in the feature file, which is what is usually used for AI/ML and later analysis.

2.1.2.6. Store and Transfer

They use campaign storage for the data they are sharing with the DSI team. The data is copied to the campaign storage by the PI. There is no real-time workflow required.

2.1.2.7. Gaps, pain points and lessons learned

As mentioned previously, this is one of many possible AI/ML workflows for radiograph data. One gap is the existence of data management infrastructure for this domain that can support more than one of these AI/ML workflows. For example, it would be useful to be able to apply some of the metadata database design used in this workflow to an active learning workflow, or to find something more general that could apply to both workflows in order to give AI/ML analysis designers support that is generally useful.

One pain point is that visual tools for analysis of metadata should ideally be programmable to incorporate different data and different datatypes, because at least now, as projects are trying new AI/ML methods and approaches, the workflows are fairly fluid and tools can become out-of-date or left behind.

The radiograph project has indicated that although exploratory, manual visualization tools for metadata analysis are useful to those new to the project or for those coming up with new approaches, automated tools that are more scalable are increasingly important. The data size and the complexity of the analysis is going beyond manual GUI use and we need to look toward more sophisticated ways of analyzing metadata from these projects. However, as mentioned before, if the metadata is available in an infrastructure that is fairly general, it should be able to support such tools.

2.1.3. Ensemble Data Management Infrastructure for AI/ML Applications

In this use case, we explore simulations supported by the Common Modeling Framework (CMF). The CMF helps simulation scientists at LANL employ best practices when designing and running simulations. It provides an infrastructure to systematically develop and run simulations as well as evaluate experimental setups. The goal of CMF is to provide a single all-encompassing structure to facilitate better decision-making for hydrodynamic codes across the lab. It is common to use the CMF to run "ensembles". Ensembles in this context are simulation runs performed where each is using a parameter set within a range.

Identifying the appropriate information to save when running CMF simulations (or any simulation) has a multitude of benefits both short-term and long-term. In the short-term, ensemble visualization techniques on simulation metadata can help users better understand and navigate their overall parameter space. In the long-term, these short-term studies can help build a better data-management infrastructure for advanced AI and ML applications, which often create training datasets that are selected from simulation ensemble run inputs and outputs, as in the radiograph example previously presented.

In many ways, ensemble analysis can be considered a first step towards AI. It forces scientists to consider what attributes, key values, images and tabular data is important to uniquely identify a simulation run. Once such decisions are made, ensemble visualization tools such as parallel coordinates plots, association metrices and clustering tools allow scientists to explore correlations between inputs and outputs. Such studies are key to determine what aspects of a simulation are most important to train an ML model and how to ensure accurate predictions on unseen data.

2.1.3.1. Plan and Design

CMF categorizes simulations based on the corresponding "authority", which restricts access to those who should have it. Example authorities include LAPA (Lagrangian Applications Project Authority) and EAPA (Eulerian Applications Project Authority). These authorities provide a general definition of input and output parameter formats for simulations supported under each authority. As part of the DSI project, we take advantage of these standardized formats to identify input and output parameters and associated values. These parameter/value tuples are extracted and saved as metadata. We also work closely with technical subject matter experts to understand and catalog additional inputs and outputs that must be extracted and saved from each particular simulation. This ensures that we capture data science infrastructure needs for CMF. This includes integration of local post-processing scripts. We also capture images, plots and other relevant information that might help identify and differentiate one simulation from another.

2.1.3.2. Acquire

CMF simulations are run on LANL HPC machines. Results are saved to designated locations on LANL networks. Sets of CMF simulations can be run jointly as a batch. Outputs range from tens of Megabytes to several Gigabytes in size.

2.1.3.3. Process and analyze

Extracted simulation-relevant information is saved using DSI architecture in a backend database. Current backend options include an SQLite database or as a csv file. The csv file can also be read into an SQLite database for processing and queries on the SQLite database can be saved out to csv format. The most common workflow is to save all parameters and values into an SQLite database, then query to extract relevant information for a specific scientific question. Results of queries can be saved out into csv format and loaded into visualization tools such as Cinema:Explorer [32] for robust exploration and analysis.

Ensemble visualization tools such as Cinema:Explorer allow for comparison between multiple simulation runs. Scientists can compare and contrast the impact of different inputs to the outputs of a simulation. In additional, scientists can apply evaluative metrics such as the Euler distance to understand which simulations are similar to one another and which ones are outliers.

An example of this framework is shown in the figure below. This is an ensemble of simulated flyer plate experiments run using CMF. A simulated beryllium flyer is hitting a stationary beryllium disc. The velocity is mapped through a simulated VISAR detector to record the impact shock. We vary various attributes of the simulation including the flyer thickness, radius, initial velocity of flyer, target thickness and target radius.

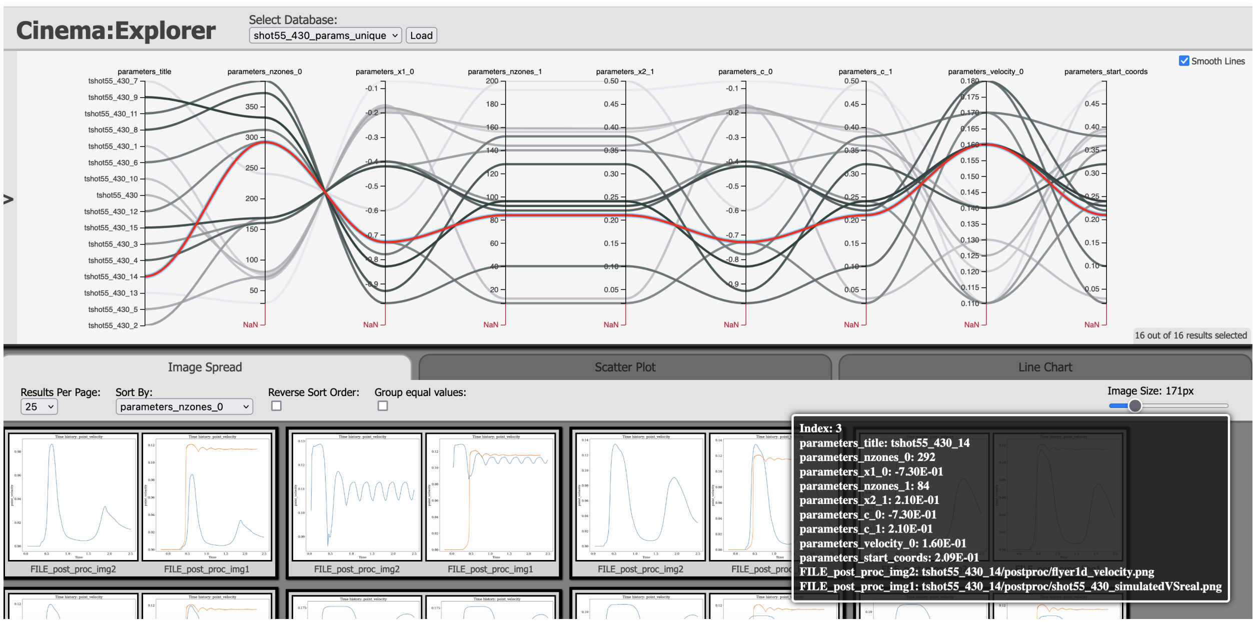


Figure 2-10. Cinema Explorer visualization of 16 shot_55_430 simulation runs: Comparative metrics applied to the numerical DSI-extracted properties of simulated experimental runs identify which runs are closest to the selection. The selection is highlighted in red. The remaining runs are colored from closest to furthest (dark gray to light gray gradient). This example uses a column-normalized Euclidean distance metric. This type of analysis can also be applied to DSI-extracted simulation information that will become part of AI/ML workflows. Image by D. Banesh.

2.1.3.4. Archive

The workflow from Figure 2-10 is often iterative. A suite of simulations are conducted, analyzed and changes made to the code within CMF. The suite is run again with the updated code and outputs are compared to the previous set of results. Currently, we are saving all outputs and iterations as is on the DSI campaign storage on multiple networks. Permissions are set to the original user who generated the code so only that person has access to the data on campaign storage. However, we are considering additional frameworks where only changes to the parameters and values are saved. Similar to a git repository where changes, additions and deletions are stored with each update, a similar format might help mitigate future space requirement issues.

2.1.3.5. Share and publish

Several aspects of the DSI infrastructure allow users to easily share DSI-related tools and databases as it relates to CMF. Storing data in SQLite and csv files make it easy to share and view metadata as it relates to CMF. In addition, front-end tools like Cinema:Explorer is integrated into the DSI architecture robustly. These tools are easy to open in any browser and self-contained. Therefore, it is easy for users to send the tools and associated data to collaborators.

2.1.3.6. Store and transfer

CMF runs are conducted on multiple networks. Group permissions restrict access to the data.

2.1.3.7. Gaps, pain points and lessons learned

A lesson learned when determining which metadata to gather from CMF simulations for ensemble analysis and ML exploration is that identifying important parameters if often an iterative process. The set of parameters a scientist identifies as vital today is likely not what they considered important two years ago and will likely change again in the next six months. Therefore, it is essential that underlying framework and front end tools are as flexible as possible to allow new parameters to be added to the database. In addition, scientists might want to generate new parameters in real-time, when exploring their data in Python or using Cinema. Tools that can easily allow for this will ensure the most accurate parameters are used for exploration.

2.1.4. Performance Application via DSI Supports Simulation and AI/ML Workflows

The DSI project capability applied to performance, testing and continuous integration/development provides data infrastructure needed for "MLOps" within AML workflows. MLOps applies DevOps methodologies to AML workflows. MLOps can include ML model registry, ML metadata store, ML pipeline automation and CI/CD pipeline orchestration.

LANL's ASC applications will typically run a range of performance and code testing or continuous integration workflows on a daily basis. By integrating with ASC tools, the DSI project can support ASC teams to automate the search and query process while respecting security requirements. An example workflow was developed using code performance and testing data from LANL's Lagrangian Applications Project (LAP). In the current workflows, LAP sends performance data to an on-premises data warehouse on a nightly cadence. Challenges in the performance data workflow include mixing POSIX and non-POSIX components, creating security concerns, and the use of proprietary tools within the data warehouse. The workflow for the code testing data is equally problematic as it relies on a nightly email that can have inconsistent formatting that is neither machine-readable nor easily digested by human recipients. Email likewise poses a potential security concern.

The DSI project has developed a workflow that captures performance and testing data in SQLite databases, where POSIX file permissions can be enforced and data can be accessed through SQL queries. This creates a machine interface where queries can be automated and subsequently integrated into more user-friendly web pages or dashboards. The resultant workflow enables access to higher quality performance and testing data with a more secure backend and more flexible and user-friendly front-end capabilities. These capabilities provide a foundation for potential future efforts to apply AI/ML approaches to these data streams.

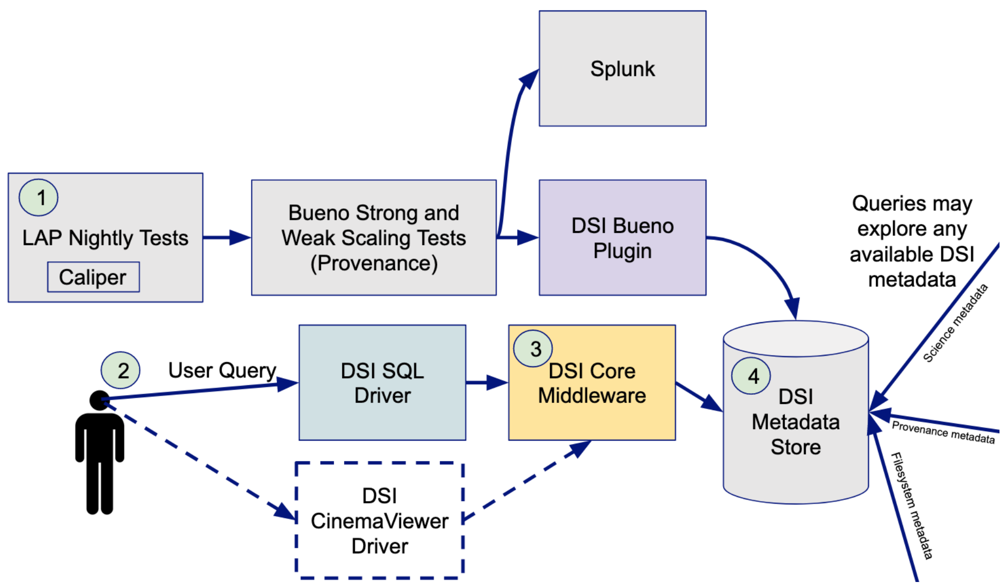


Figure 2-11. The LANL performance application workflow above shows the following steps: 1) LAP nightly testing via Caliper and Bueno sends data to DSI back-end, 2) user enters SQL query text string into DSI Python API, or alternatively does a visual query with Cinema, 3) the DSI API query fetches data according to the query, and 4) the DSI SQLite back end accesses a single file to query the data (with a stretch goal to merge multiple files to provide data for a query eventually). Image by Q. Wofford.

2.1.4.1. Acquire

The LANL LAP project uses the LLNL performance analysis toolkit, Caliper [33], for code testing. In-house, LAP uses the LANL performance engineering framework, PerfEng, to submit the performance testing and uses the open-source reproducibility framework, Bueno [34], to package the Caliper test outputs. These outputs are sent to the Splunk [35] log analysis tool. Integrating Bueno with the DSI API enables the performance and testing data to be stored in the DSI databases. The workflow is illustrated in Figure 2-11.

2.1.4.2. Process and analyze

The original workflow uses application-specific databases with high maintenance needs. The ongoing DSI effort is making the performance data accessible through the Python SQL interface across all DSI-integrated LANL ASC codes. The future goal is to support AI/ML tools by leveraging DSI to search/query all relevant ASC program outputs. This will be particularly valuable in validating (or invalidating) training sets. By including performance and testing data workflows, reproducibility and provenance are inherently part of the AI/ML workflows.

2.1.4.3. Archive

No plans to archive data yet.

2.1.4.4. Share and publish

The reliance on POSIX controls within DSI infrastructure is sufficient to meet the sharing needs.

2.1.4.5. Store and transfer

Performance and testing data will be conducted on multiple networks. Group permissions restrict access to the data.

2.1.4.6. Gaps, pain points and lessons learned

One gap/requirement that was exposed by the performance use case is the need to support queries which read and write sensitive LANL mission data. This is a capability that carries high risk. No user has access to exactly the same information, and information is spread across many files. A useful query result presents information from many files granted by different permissions authorities. Queries must not be saved with a single permissions authority, or information could be leaked and query power is reduced. The DSI tool must allow users to view and save useful query results, without mixing the permissions authorities of the source information. A formal algorithm is required to suit the user capability demands without sacrificing security compliance. Efforts are ongoing and this requirement spans multiple use cases.

2.2. Lawrence Livermore National Laboratory

2.2.1. Overview

LLNL identified 3 representative use cases. The first case is the Inertial Confinement on El Capitan (ICECap) workflow and the Inertial Confinement Fusion (ICF) pre- and post-shot analysis workflows at large, the second one is Climate Modeling community effort, and the third one is an emerging type of work: edge computing to drive real-time decision making. Together, these 3 projects cover a large spectrum of LLNL's mission and highlight various successes and challenges.

2.2.2. Inertial Confinement on El Capitan (ICECap)

Designers decide which shot will be fired at the National Ignition Facility (NIF) through analysis of the results of many "pre-shot" simulations. Similarly post-shot simulations aiming at reproducing an experimental shot are run.

The Inertial Confinement on El Capitan (ICECap) effort aims at demonstrating how LLNL's upcoming exascale machine (El Capitan) can employ AI/ML to explore a large design space for Inertial Confinement Fusion (ICF) experiments using multi-fidelity calculations and high-fidelity surrogate models.

2.2.2.1. Plan and design

When preparing a pre-shot simulation designers look at a variety of parameters related to the size and materials of the target and variations in the laser pulse. After the experiment is completed, "post-shot" simulations and analysis are used to identify uncertainty and refine simulations.

For ICECap, the AI-driven optimization workflow is allowed to freely explore design space, given a set of parameters that can be varied. Although this allows the AI engine to find potentially unexplored areas of design space, it adds complexity and robustness requirements to the workflow.

2.2.2.2. Acquire

In the current ICF workflow, pre-shot simulations are used to design an upcoming experiment and predict its anticipated results. A variety of simulations are run to determine the geometry and materials for both the capsule and the hohlraum and the laser configuration (power history, beam direction, and wavelengths). Hohlraum and capsule simulations are run separately and in an integrated fashion. The hohlraum and laser configuration are optimized with a nominal capsule and the laser energy is sourced into a capsule-only simulation to optimize the capsule. The fully integrated hohlraum and capsule simulations are expensive and less common during these iterations.

Currently, this is done via hand-editing shell and Python scripts. Ensembles are run using workflow orchestration tools, such as Merlin [36]. There is a current effort to standardize this via template files. These templates are sent to the NIF File Input Editor (NIFE) via a command line interface or configuration files. NIFE is a collection of loader/writer in iDesign (see bellow) that writes other input files. The ICF Program has a "How Things Should Look" group working to standardize this process.

A post-shot simulation aims at reproducing an experimental shot. The first phase in preparing for such a simulation is to gather the input files for the "deck" (input file used to drive a physics simulation). Data related to the specific experiment, such as target, topology, geometry, moisture content, laser pulse, etc., is pulled automatically from various servers. Scripts pull this data and generate inputs. These scripts are part of an integrated python environment developed by LLNL's ICF Program: iDesign[2]. These inputs are then passed to "target builder" which produces the input files for the code. This process is still very "patchy", with required human interventions in the loop, but progress is being made toward full automation. It is worth noting that not all the data is easily accessible or accurate. There is still the needs for a user to proofread the files.

ICECap intends to revolutionize this process by optimizing pre-shot design using AI-driven, multi-fidelity design optimization and atomic physics ML surrogates to accelerate simulations. This adds significant complexity to data management, including comparing inference queries compared to training data and logging real-time queries for retraining. This must all be handled during ensemble runs, with versioned models for reproducibility.

2.2.2.3. Process and analyze

For current ICF workflows, data needed for uncertainty quantification (UQ) or shot tuning is not available via a database. It can be available in a various, complicated, and not automated number of ways such as notebooks, email, MATLAB scripts ran manually, etc. The data currently available (laser pulse information, experimental archive, target specifications, etc.) spans between six and a dozen different databases that are not linked. Some of them are not managed by LLNL, such as General Atomics target data. The point of iDesign is to help present all of this in a coherent format through a single Python script. The input to the simulation is then standardized into a set of files.

The ICF Program recognizes that systems for pre- and post-shot analysis are constantly evolving and so must the data management and infrastructure. The program has allocated more resources to this effort, yet there is difficulty in implementing systems to support a variety of scientists' needs and data archiving policies.

Figure 2-12 illustrates the integration of software and hardware components required to execute these AI-enhanced physics simulation workflows on El Capitan. ICECap utilizes Merlin and Maestro for workflow orchestration and Kosh and Sina for data management. Inference queries that are determined to be out-of-scope for the atomic physics surrogate are output to files during the simulations. While the simulation ensembles continue to run, another job adds these

[2] iDesign: https://pcmdi.llnl.gov/https://rzlc.llnl.gov/gitlab/idesign

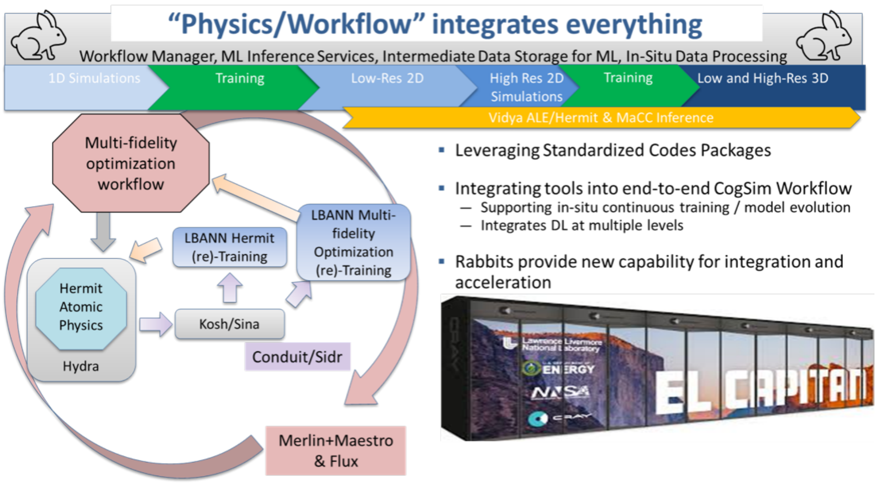


Figure 2-12. Physics workflows on El Capitan.

out-of-scope queries to Kosh and retrains the model when there is sufficient data for retraining. Data is annotated in Kosh to indicate that it was used for training that specific version of the surrogate model. The surrogate models are versioned for reproducibility.

2.2.2.4. Archive

The current ICF shot data pipeline puts everything into a database, essentially a pandas dataframe. Each shot represents a few gigabytes worth of interesting data, adding up to a terabyte scale for all of the NIF shots that exist today. The iDesign framework needs to accommodate changes in the underlying data sources over time. Improved documentation in the data sources would be very beneficial to both developers and users. Data should be self-describing (e.g., units are recorded along values).

Post-shot ensembles are catalogued, analyzed, and archived. There is an effort to better organize what is archived. Not all the data can be kept for large ensembles. Raw data is discarded or used in-situ (e.g, x-ray diagnostics), without a lasting archive after processing due to storage limitations. In contrast raw data that is written to files is kept for pre-shot runs for some time, facilitating generation of visualizations or deep dive detailed analysis. Note that simulation data is written to files for a small fraction of the timesteps in a simulation, so most of the simulation data is lost during the simulation. Pre-shot simulation data may be discarded after the shot depending on the designer. Ideally the input files are archived along with the simulation data, but this is not enforced. The process of archiving simulation data tends to be manual making it hard at times to know the provenance. Standardization of archiving inputs/outputs would help both by ensuring that archives exist and providing common information about provenance.

The length of time the data is kept depends on each user preferences and quotas. Standardizing handling of simulation data, including inputs, would enable longevity and deeper analysis across a larger variety of experiments.

There are no current plans to archive data for ICECap, although that will need to be handled if this AI-driven design optimization is used more widely by the ICF program. Interrogation and reproducibility of these results will be of utmost importance, because we expect AI driven designs to venture outside of previously explored areas. This will result in the need to deeply analyze and understand the predicted designs.

2.2.2.5. Gaps, pain points and lessons learned

Input decks for current ICF work continue to grow in complexity increasing the desire for standardization. There is an effort to standardize these by using version controlled common models which include standardized output and post-processing tools. This is ongoing with various degrees of adoption within the community. Meanwhile, data per shot are expected to grow by 10% to 20% per year.

Designers can become more productive and gain insights by having more extensive and easy access to data. Although sophisticated data systems exist for NIF experiments, they often facilitate looking at the detail of a single experiment, while designers need to perform analysis across a broad variety of experiments and simulations, so they have developed their own methods for this analysis. This limits knowledge of how data were generated and confidence that it meets accuracy requirements. Building tools to automatically archive standard data with consistent metadata would allow designers to focus more on design and analysis than software engineering.

ICECap represents a large-scale AI/ML effort, demonstrating the need to better standardize and archive inputs and outputs for multi-fidelity simulations. Although this effort focuses entirely on simulations, it can also help identify requirements for pre- and post-shot analysis as described above. Requirements for consistent data generated from a large variety of ensembles for real-time training of AI/ML models necessitate standards and automation. Provenance is also an inherent part of this project. Whatever designs are identified during AI-driven design optimization will require deep, high-fidelity analysis, so it is important to know how results were generated using the AI/ML model. Complex workflows are needed to run simulations, process data, and retrain models without ever blocking compute. These workflows exist at the outer loop, driving the ensemble settings, and the inner loop, logging atomic physics results for retraining. This effort will continue to inform infrastructure needs.

2.2.3. Climate Data – Earth System Federation Grid

Started in 1989, LLNL's Program for Climate Model Diagnostics and Intercomparison (PCMDI)[3] successfully led the effort of organizing the international climate community which allowed tremendous advancements in the domain's science and data exchange technologies. The community assembled to run Model Intercomparison Projects (MIP). The first one was in 1989, when various centers around the world agreed to run their climate models with the same sets of input conditions. Many technical hurdles were encountered (data format, data access, data

[3] Program for Climate Model Diagnostics: https://pcmdi.llnl.gov/

storage, etc.) but the result was a resounding success, for example it demonstrated that ALL models had a systematic cold bias around 200hPa. A fact that most modelers knew but thought was specific to their individual models. This led to drastic improvements in models and to a strong engagement from the community thereafter. Expanding the concept to other MIP (ocean, coupled, etc.) PCMDI organized the community to support experiments, design, and data sharing. This led to dramatic improvements to the models in general and enabled policy makers.

2.2.3.1. Plan and design

Determine research goals, such as what kind of model are we comparing (Atmosphere vs Ocean vs Coupled, etc.) and methods (which experiments will be run, determine input data); identify required data infrastructure and create data management plan.

2.2.3.2. Acquire

Each center runs the simulation. Originally, they would send the data as-is to PCMDI in whatever format their code would produce. Eventually tools were developed to help rewrite the data to be published according to standards and conventions.

2.2.3.3. Process and analyze

As software was developed to rewrite the data, a few standard outputs emerged, enabling easier comparisons. Eventually conventions were established regarding the format and the metadata. Climate Forecast conventions are key to collaborations successes, by standardizing names and metadata. These conventions are the result of many years of incremental progress and a cultural buy-in.

Comparison with experimental data (instrumental and/or satellite) is facilitated by rewriting them to a matching format and structure.

2.2.3.4. Archive

PCDMI used to be the central archival point, with a few copies mirrored elsewhere for easier access.

When assessing what data to save, there are many considerations including, what data is easily reproducible (time vs. hardware cost), what goes on tape (causing potentially long delays in access), and what stays readily accessible. Maintaining the archive proved to be crucial point as well, once enough hardware is procured there is a need to maintain it up and running, recover from crashes and maintain security.

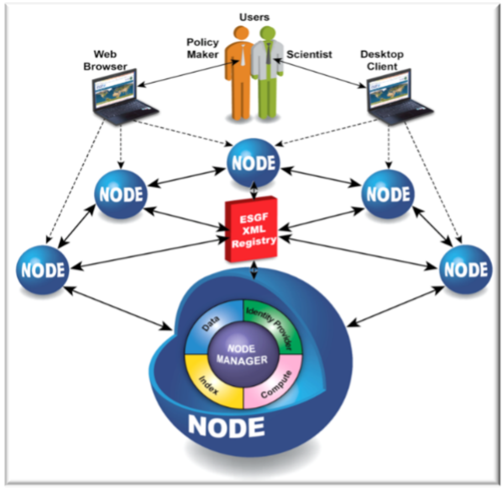


Figure 2-13. LLNL's Earth System Grid Federation allowed collaborating centers to publish and share data across the globe.

2.2.3.5. Share and publish

Centers would download the data from PCMDI. At some point the amount of data to transfer became too much to handle and the Earth System Federation Grid was born, allowing centers to leave the data where they were produced but making them accessible to all.

Climate Model Intercomparison (CMIP) data is federated over 30 Earth System Grid Federation (ESGF) nodes. Data is three dimensional plus time, stored in NetCDF5, structured via filenames and directories. This helps with data discovery.

The Earth System Grid Federation (ESGF), illustrated in Figure 2-13 allows research and non-research institutions to share large volumes of curated data. Major investments were made in regards to data transfer and authentication, the community relies heavily on Globus to solve these problems, routinely registering 1000s of MBps transfer rates (https://dashboard.globus.org/esgf/) . The MIPs led to documents such as the International Panel on Climate Change (IPCC) reports (Figure 2-14), which are the basis for many international policies on climate regulations. The Second Assessment Record of the Intergovernmental Panel on Climate Change was the first document to officially acknowledge a discernable anthropogenic impact on climate change. LLNL's Program for Climate Models Diagnosis and Intercomparison was one the major science drivers behind the report.

2.2.3.6. Store and transfer

Data stays were ti was generated. Some core set gets replicated at multiple "central" sites (such as LLNL). Data is downloaded from the generating center or in some cases processed there and a substet is downloaded.

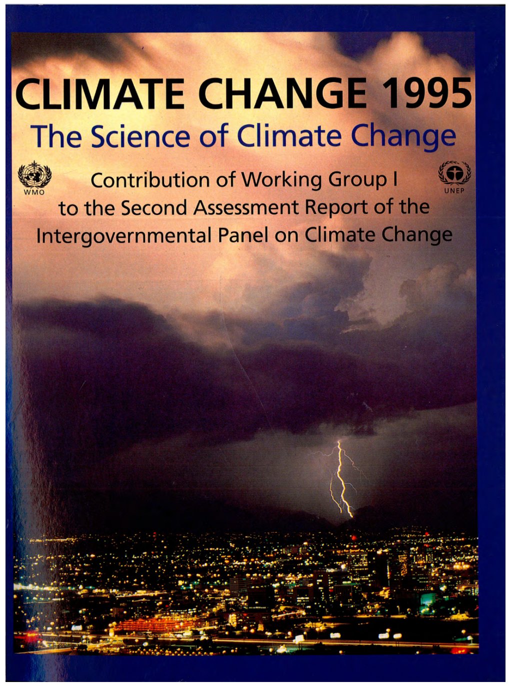


Figure 2-14. The Second Assessment Record of the Intergovernmental Panel on Climate Change was the first document to officially acknowledge a discernable anthropogenic impact on climate change. LLNL's program for Climate Models Diagnostics and Intercomparison was one of the major science drivers behind the report.

2.2.3.7. Gaps, pain points and lessons learned

While this example is not specifically about ML, it demonstrates best practices when dealing with large amount of data coming from many different institutions. This community also recognizes that AI/ML techniques and technologies can be incorporated into their efforts.

Cloud: There are many communities accessing this data: climate modeling, impact groups (behind technologically) and startups. This can lead to still very heterogeneous data. As more and more startups are accessing the data, there seems to be pressure to move away from NetCDF and toward cloud storage. More and more computations are made next to the data to reduce data movement (and caching, etc.). A subset (3Pb) was sent to Google Storage where startups are using it. Interaction with other constantly generated sources is shaping up (health, energy, planes, soil, etc.).

Reproducibility: This is a big problem, how much provenance do you keep? Generally, it is considered reproducible as long as the hardware does not change. Some systems keep track of versions for the whole software stack.

Storage cost: For reference CMIP3 was 70Tb, CMIP5 2Pb and CMIP6 which just started is already at 14Pb with $14M in hardware. Cloud solutions offer "free" storage but "pay-per-use" when downloading the data. Computing in the cloud adds additional cost.

Maintenance cost: Hardware and software as well as security need constant maintenance adding to long-term costs that are significant.

Disrupting technologies:

The field is looking into earth digital twins, which is a disrupting factor because the amount of data this requires is not manageable at this time, as AI is also ramping up as fast and flexible solutions. Such twins could generate Pb worth of data for 1 hour of simulation. This data needs to be reduced somehow to around a Pb for storage (reduced frequency, reduced number of variables stored, etc…). There are also considerations of edge network because earth digital twins have a high degree of interaction with externally generated source (from energy domain for example) and result in a combination of satellites data, AI surrogate and prediction data. We're seeing a fusion of AI, observation, simulation and secondary generated data sources.

2.2.4. Edge Computing and Real-Time Decision Making

Livermore is investing strategically in multiple edge-computing use cases. These require new types of workflows where computational resources interact with physical devices in real time. Understanding complicated physical interactions, such as those in high energy density plasmas, would benefit greatly from higher frequency experiments. This example has pushed the exploration of high rep-rate laser facilities, allowing hundreds of experiments per minute instead of 10 experiments per day. Edge computing coupled to fast surrogate models could help analyze and adjust facility functioning in real-time. We refer to this as a "self-driving facility". Another use case for a self-driving facility using edge computing to make real-time decisions is advanced manufacturing.

2.2.4.1. Plan and design

The high rep-rate project is using simulation data generated on HPC systems to train the ML models. The models are used in inference mode only on the edge. The ML models are used for inference because they emphasize high throughput over accuracy. High throughput is accomplished both through low-latency for the hardware and low computational cost of these models. This facilitates real-time analysis of data acquired through sensors to steer future simulations. Higher resolution simulations can continue to run outside of the edge (potentially on a HPC system) to improve the models which are in turn sent back to the edge once improved. The project's goal is to use distributed object storage to maintain the connection between continuously generated high-resolution simulation data and the surrogate models that are used on the edge.

2.2.4.2. Acquire

Data exists at multiple levels for this project:

- High resolution/high-volume simulations generate raw data on HPC systems.

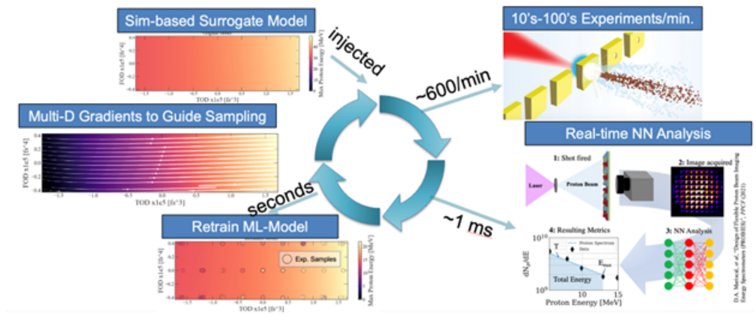


Figure 2-15. AI-driven experiments will be transformative for laser-driven HED physics exploration & optimization

- A subset of the simulation data relevant to the surrogate model is moved to laptops train and retrain models.
- The model is deployed at the edge (near experiments) for real-time inference.
- Sensor data from the experiments are run through loalized diagnostic models for real-time inference.
- Analyzed data (scalars) are fed back to physics surrogates for retraining of simulation-based models and new driving commands are sent to experiment

Currently the handoff of data between all of these systems is manual because there are no good existing workflow tools available to them.

2.2.4.3. Process and analyze

While running the experiment and processing data on the edge there might be a realization that the model needs to be re-trained, because the model may need higher accuracy in the current experimental state. Similarly, some experiment might need to be run again because a better model has been generated since it was originally run. This requires sensor data to be passed to the HPC system to determine what high-resolution simulations need to be run for validation. The simulation output then needs to be processed and used to train the model on laptops and the trained model is then moved to the edge to be run in real time. Figures 2-15 and 2-16 show illustrations of AI-driven experiments and the required infrastructure. This project has proposed the use of a distributed data store with connections to all of this data and the cloud to better streamline the transfer of information.

2.2.4.4. Archive and Storage

Each experiment has a relatively small data print, about 50Mb worth of images and oscilloscope signals and scalars, but because these are high throughput experiments (run at 10Hertz) we need about half a Gb worth of storage per second which is later sent to long term storage. There is currently no key-in solution for such storage needs (both for capacity and movement).

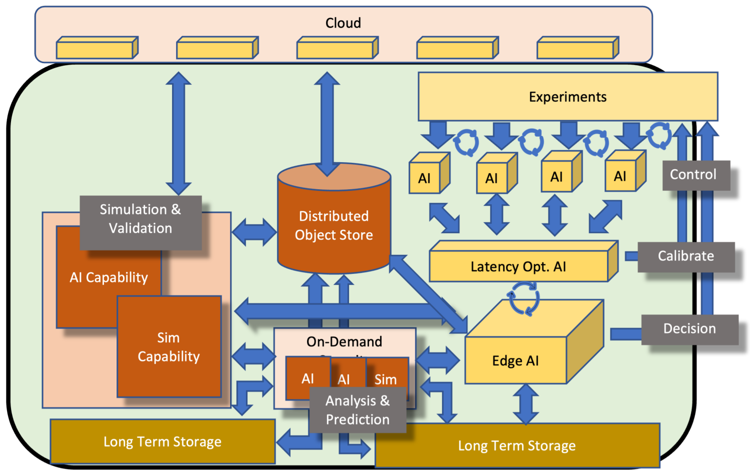


Figure 2-16. Infrastructure for AI-driven workflows

2.2.4.5. Gaps, pain points and lessons learned

This project has identified many gaps with the current infrastructure and has developed a graphical representation of their ideal data infrastructure, incorporating cloud services and a distributed data store. An important aspect of this infrastructure is that it links many sources of data for automatics training and retraining ML models that can be used with sensor data at the edge for real-time decision making that improves with more data.

The main pain points identified by this project are:

- Real-time AI decision making can have a high consequence, because a human cannot be in the loop to finalize any decisions made. This is a use case that requires safeguards to ensure that low quality inference is not able to damage a facility or cause injury.
- Data movement increases vulnerabilities because data corruption or interception becomes more likely. This requires additional checks on the quality of the data when it is relocated and security to ensure that improper access does not occur.
- Edge computing requires low power consumption, which impacts the complexity and throughput of the models that can be processed on them.
- The technology is still emerging, and no good workflow tools exists especially when it comes to handling steps at very different time rates.

2.3. Sandia National Laboratories

2.3.1. Overview

The examples selected from Sandia represent a broad set of problems relevant to Sandia's role in the Advanced Simulation and Computing program. These include:

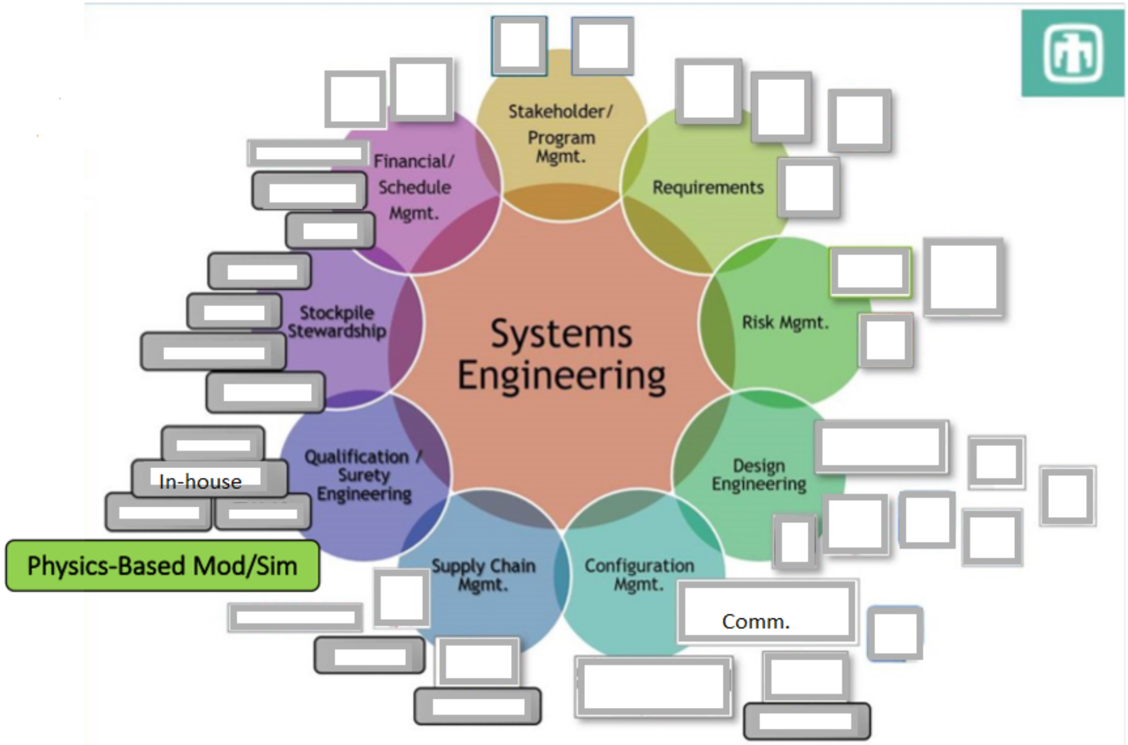


Figure 2-17. Sandia uses over 200 different (when counting 228 ASC-developed tools as *one*) commercial and in-house software tools in design workflows for nuclear deterrence.

- Physics-informed machine learning to support advanced manufacturing and modeling of as-measured parts using neural-networks derived from ASC simulation outputs.
- Automated data exploration and behavior discovery for improving use of HPC systems and applications.
- Creation of models based on neural-networks derived from HPC simulations for real-time wildfire prediction and response management.

Sandia is engaged in building a Digital Engineering Ecosystem (DEE), in which ASC modeling and simulation tools play many critical roles. AI/ML-based design screening models and high-fidelity multi-physics models are key aspects of the Accelerated Digital Engineering (ADE) effort at Sandia. The use cases examined here use ad hoc methods (or no method as yet) to capture and reuse data ontology and provenance information. Also common to these cases is that users expect multiple changes of both the workflows and the structure of data to be managed as internal and external project requirements shift. The staggering complexity [37] of the DEE (see Figure 2-17), the multi-decade lifecycle for SNL computational models, and the scarcity of people with the depth of knowledge needed to use and update the models as all the supporting hardware, software, standards, and requirements evolve means that modeling and simulation will require changes to computing, network, and storage architectures and to ASC codes at Sandia.

Maintaining and updating the AML models derived from experimental or surveillance data filtered and expanded through the lens of HPC modeling and simulation will require that:

- We need to maximize automation of model testing and model translation among tools in the modeling workflows wherever possible, so ASC HPC tools and clusters must be fitted for:
  - automated use.
  - automated provenance capture and retention.
  - automated data translation to adjacent domains or tools with distinct information schemas.
- We need adoption of common (to the nuclear security enterprise (NSE)) and standards-oriented (to the commercial modeling tool providers) frameworks for managing our model data to enable thes automations. The information capture and sharing tools adopted must span the ML researchers' laptops to the HPC simulation environments to the design engineers to the surveillance teams.
- We need multi-decade (essentially permanent), fast-access storage for automated periodic (daily to weekly) verification that curated HPC models continue to function with their reference data sets and reference workflows as all parts of the DEE environment evolve. This also supports periodic retraining of ML models as ML algorithms evolve and new training data or testing data becomes available.
- We must have career incentives for expert staff to perform the unglamourous (even with automated support) tasks of digital model capture, maintenance, testing, and updating as the environments evolve.

Integration of AML approaches into Sandia DEE workflows leveraging HPC is very like integrating any other engineering or science (EngSci) software tool to a larger workflow system for use in a highly traceable, reproducible way. The owner and maintainers of each EngSci tool are the ones best positioned (and often the only ones positioned) to create the needed new ontology description or map of existing ontologies for their tool, enabling automated data transformations and data discovery; this is especially true for ASC-developed codes that will be integrated into highly reproducible, highly documented AML-enabled or digital engineering workflows.

Common to all these engineering tools, HPC and not, are a large, deep, and complex semantics captured in each software; in every case, the semantics of the tool partially overlaps a few of the other tools. In Sandia's digital engineering strategy, the thread of information linking results to their authoritative antecedents is enabled (in the best case) by an ontology-based information broker. The broker is EngSci-vendor neutral, depending on domain experts (in most cases, the companies that provide the commercial tools) to define and maintain the descriptions of data objects, fields, and relationships needed to support automated mapping of data from one EngSci tool to another (using the tool or domain-specific ontologies) during workflow execution.

The report "ASC Credibility for Scientific Machine Learning Assessments" (Acquesta, et al, pending release) defines numerous examples of how much more metadata there is for scientific

ML and how complex gathering it is (multiple human sources, with different tooling needs) as metadata accumulates around the data and derived data traveling through workflows. That report scope excludes infrastructure needs other than:

- Having the ability to reproduce results including estimates of uncertainties.
- Having the ability to capture/store/access metadata throughout the lifecycle.
- Policies being in place to ensure these abilities are applied in all necessary institutional and enterprise contexts.

2.3.2. Physics-informed Machine Learning Material Models for Solid Mechanics

The overall project goal is to collect imaging data, convert it for use in high-fidelity material behavior simulations, generate simulation results from many application (load) scenarios, and then train a neural network to more quickly reproduce the material behaviors seen on relevant scales. This neural network model is then used for production simulation work, drastically reducing 3d simulation time. The ultimate aim is to enable workflows which support certification of additively manufactured parts for the stockpile using diverse data types and data sources. This approach (using ML to correct model-form error) addresses sparseness of experimental data and model-form error to accelerate modeling of materials such as polymer foams and additively manufactured metals [38, 39].

2.3.2.1. Plan and design

The material modeling team's workflow was initially manual, making data loss risk due to personnel changes very high. A related project has since developed a set of tools adopted by this team, Experiment Tracker (ET), to maintain the various data files with a metadata infrastructure and a web interface to a shared file system suitable for use by both experimentalists and modelers. The users and developers of ET see it as a temporary solution. ET does not automatically track the relationships among derived models and simulation results, nor among a set of closely related experiments (such as CT scans of a set of "identically produced" additively manufactured samples).

2.3.2.2. Acquire

Acquiring data from the advanced-manufacturing imaging facility is a multi-step manual activity. It first requires the researcher to manually extract terabytes of 2D and 3D raw imaging data and metadata (csv or spreadsheet) of additive-manufactured metal samples from the imaging system. Since the imaging systems do not share a network with the computing platforms used for analysis, the researcher must transfer hard drives (or some other storage media) to a networked workstation. Next, the researcher uses standard file-transfer tools (e.g., FTP) to move the data from the workstation to NFS-mounted storage accessible to HPC platforms. Once the data is on project storage, the reseacher needs to ingest imaging data and metadata into the Experiment

Table 2-1. Data Software Stack for Physics-Informed ML

Analysis & visualization tools	Pytorch, Jax, python, C++, julia, Sierra Solid Mechanics, ML research codes
Data interface tools/middleware	Experiment Tracker web and python UIs; manual transfers (FrETT, rsync); file formats (CSV, MS Excel, Exodus, Numpy)
Data services	README files, file hierarchy conventions; Corporate mongo database Data architecture & interoperability frameworks
Data architecture & interoperability frameworks	Experiment Tracker architecture; no interoperability framework
Storage, networking, and computing systems	NFS; Linux; 1Gb network; Engineering servers, HPC clusters
Org. support, culture, & workforce	University researchers & ASC funded staff develop analysis methods and data sources. Long-term data curation and sharing are not supported as separate activities. LDRD grand challenge funded developing of Experiment Tracker

Tracker (ET) tool. Care must be taken to avoid associating the data with other information which may render the whole classified. Experimental data cannot be ingested into ET without proper metadata being provided. Key features of ET in the users' view are:

- A utility included that generates unique identifiers that can be bar-coded and applied to samples measured or to the master media of a raw data set.
- A design that is flexible with regard to the data schema, which varies frequently across experiments.
- Supports managing groups of hundreds to thousands of files, small to very large, for a single experiment.

Experimentalists typically rely on an informal "data librarian" (an intern or staff from the HPC simulation team) to ensure the correct metadata and data is collected for each experiment.

2.3.2.3. Process and analyze

Before analysis, the raw data is processed to filter out hardware biases from the imager, and used to generate a finite-element geometric and material representations for use in high-fidelity material behavior simulations. Both of these steps use software that evolves rapidly, making it difficult to consistently reproduce derived results. Partial automation of the workflows in this analysis has been completed, in particular there is an automated workflow (home-grown script) that performs early-stage filtration and generates initial mesh representations of the experimental data.

Prototype software integration between the solid mechanics model and third-party GPU machine learning code (PyTorch) makes the processing (both training and resulting ML model deployment) brittle. The developers are looking for a more sustainable solution, where control does not flow into and out of a python interpreter from the mainly C++ code.

2.3.2.4. Archive

The raw data, high fidelity simulation data derived from it, and resulting neural network models must all be reliably archived, findable, and retrievable to guarantee future availability. Presently, the filtered data and derived data and models are all stored in project directories on low performance but widely accessible file systems, with ad hoc pushes to tape as needed to stay within storage volume quotas.

2.3.2.5. Share and publish

The resulting material models must be shared via integration into the production solid mechanics codes used in the laboratories. Integration of the derived models with existing material modeling data sources such as Granta is an open question. Wider sharing of the experimental data and derived material models is highly dependent on program requirements. Certain cases are used as public modeling challenge problems for material behaviors.

ET is designed to accommodate a central index of available data sets, but is often deployed as independent instances, eliminating the findability of data.

2.3.2.6. Store and transfer

As described in Section 2.3.2.3, raw data of the TB scale (for an individual experiment) is manually transferred (drive swaps) from imagers to equipment on the required corporate networks. For a single material model, the total raw data is currently of scale 1-10 terabytes and the output for corresponding high-fidelity runs modeling dynamic behavior is 10s of terabytes. GPU memory sizes are currently limited to the RAM of a single (100GB) device. CPU memory sizes on HPC clusters are too small for training with non-distributed algorithms; as a result, scarce large memory workstations rather than HPC clusters are used.

We anticipate a significant increase in storage capacity requirements for experiments integrated into a a future production digital engineering capability. The number of materials to be modeled is at least an order of magnitude (10x) larger. Image data resolution is expected to improve over the next decade by an uncertain factor, R, increasing the volume of raw data per image curated by R^3. The type and number parts to be imaged for material modeling purposes will reasonably be expected to expand by a factor, T. The addition of time-series imagery captured during part construction expands the data volume by the number of frames captured per part (at least thousands). Even modest values of R (3) and T (100) without time-series data lead to a need for active storage capacity of 10s of PB for data with automatic, periodic analysis as the methods, software, and platforms evolve. Similarly, single-GPU and single-node memory capacity is not expected to expand indefinitely, so distributed training algorithms capable of handling petabytes of data will be needed.

In addition to storage, we expect more significant burden on the network. Currently data (thousands of large and small files) is transferred among HPC and corporate file systems with the ASC FrETT utility [40] and custom scripts. Tracking is with the stop-gap tool Experiment Tracker.

2.3.2.7. Gaps, pain points and lessons learned

Table 2-2 lists resource areas and gaps important to the physics informed machine learning project. The most notable lessons learned are:

- Access to mass data through 1Gb pipes is limiting overall efficiency of the machine learning processes, but ubiquitous access is considered more critical to overall workflow than hardware use efficiency.
- Previously existing data and metadata management solutions were all found insufficient due to gaps in cost, UX, access ubiquitousness, or data definition flexibility. This lead to adoption of ET as a customizable partial solution.

2.3.3. HPC Resource Management and Monitoring

Sandia HPC center monitoring continuously collects petabytes per year of machine and application performance data, enabling load and performance analysis at scopes up to center-wide [41]. The collection and analysis processes are seen in Figure 2-18. Additional workflows not shown encompass the development of new analysis methods, new rendering methods (dashboards), and new data sources. Machine learning is used to model application behavior and to sort time series signals into similar groups. Table 2-3 enumerates the data technology stack currently in use.

2.3.3.1. Plan and design

The SNL HPC monitoring development and deployment team is focused on improving system performance and positively impacting user productivity by collecting, analyzing, and sharing performance data. There is no formally documented data management architecture or data ontology; rather, the team is charged with the objective of creating and maintaining a unified hardware and software infrastructure supporting:

Table 2-2. Technology Gaps for Physics-Informed ML

Resource	Current Use	Gaps
10/100Gb networking	HPC simulation	Missing direct 10/100Gb network access to ET data from HPC cluster compute nodes and data-center-hosted workstations (DCW) for ML analyses
1Gb networking	Engineering server & DCW access ET shared data	none
Database service	Index for known data sets	FAIR principles not supported
Parallel file transfer	Move large count, large file sets	Ease of use (restart after failure)
GPU-based libraries	Small ML training, · ASC simulation code (pending)	(1) Distributed-memory versions of CPU and GPU-based CT data preparation. (2) Distributed-memory versions of CPU and GPU-based training (learning) algorithms. (3) GPU with 100s of GB capacity for high precision 3dfft
NFS	ET shared data storage	(1) Long-term capacity (3d time-series of additive manufacturing process experiments) (2) Bandwidth (hours loading data)
Lustre	Simulation input, output	(1) 100Gb connectivity to engineering compute servers, GPU clusters, experimentalist data acquisition stations (2) Capacity (3) Robust long-term storage
Data interoperability framework	None	(1) No staff available to develop custom solutions. (2) No sustainable, corporate-supported solution available in HPC Linux and Linux DCW environments for petabyte data interoperability. (3) No corporate standard for tagging and supporting database to manage the identities of experimental equipment pieces and the data sets derived from each piece of equipment.
Authentication & authorization	Access to web services and files	None
HPC operating systems	TOSS3 (TOSS4 pending)	Modern open-source, high-performance ML toolkits are rarely immediately available and often unsupported altogether on outdated operating systems.
HPC compute	Solid mechanics simulation	Large gpu memory to handle 8 billion voxel single-image data (2000^3) and related FFTs

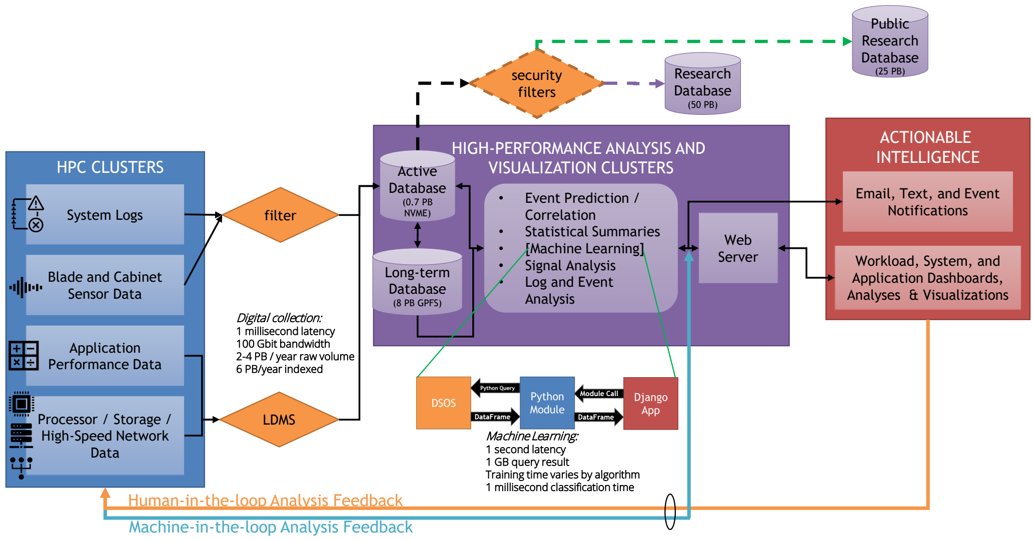


Figure 2-18. Sandia HPC monitoring data flows.

- An HPC hardware environment where new systems or data sources come along every year, while large cluster systems may persist and need new diagnostics developed for a decade or more.
- Data collection and analyses for automatically identifying and alerting appropriate personnel on the recurrence of abnormal conditions that have been previously identified.
- Data retention and analysis over the entire life of each system for crash development of diagnostics as new problems emerge.
- Understanding of aggregate workloads as a whole and by program, to enable management priority and procurement decisions.
- Understanding individual workloads to enable software developers, users, and administrators to make best use of the available systems.

As a result, data descriptions can largely be reconstructed from database table queries, and the team tactically acquires storage, processing, and networking infrastructure.

2.3.3.2. Acquire

Monitoring data is acquired from a regularly growing set of sources: messages sent from EngSci applications to the monitoring system as they execute, messages from the cluster scheduling and resource management software, operating system logs, hardware performance metrics. All of it is stored to high performance databases (0.7 PB) for immediate reuse, though data older than a few weeks is migrated to larger, longer-retrieval databases (8PB).

Table 2-3. Data Software Stack for HPC Resource Monitoring

Analysis & visualization tools	Grafana, python, pandas, jupyter, gnuplot, MS excel, time-series ML research codes for unsupervised and unsupervised classification and supervised anomaly detection
Data interface tools/middleware	Grafana data source API, Frett parallel file transfer, custom bash and python scripts, manual transfers
Data services	LDAP, apache, django, LDMS [42] data collection, SLURM accounting data, profilers (Darshan, Caliper, kokkos), file hierarchy conventions
Data architecture & interoperability frameworks	None
Storage, networking, and computing systems	User-facing: Lustre; GPFS; NFS; Linux; 1/10/100Gb networks; HPC clusters; desktops; Administrative: 1/10/100Gb networks; SOSdb database; co-hosted high rate, low latency storage and analysis servers
Org. support, culture, & workforce	University researchers & ASC funded staff develop analysis methods and data sources. Long-term data curation and sharing are not supported as separate activities.

2.3.3.3. Process and analyze

Monitoring data is analyzed with open-source database, web, python, and C/C++ tools, with the goal of producing insights and actionable intelligence. The web tools include the grafana visualization server and Apache server. Python and underlying C/C++ libraries are used to code custom analyses, including machine-learning algorithms processing large volumes of time-series data. Many, but not all, of the machine-learning algorithms are best suited to GPU-based implementations, though GPUs are not currently available to the monitoring team.

Analysis algorithms are under continuous development as the available data, intelligence needs, and available libraries evolve; the rate of analysis development is strongly limited by the scarcity of staff jointly knowledgeable in HPC performance, related numerical methods, and python coding. To partially address this gap, student researchers are employed to develop algorithms in key areas under the guidance of staff. Providing adequate data to these researchers is an ongoing challenge, which is described further in the “Share and Publish” subsection below.

Currently, the high-performance databases are co-located with analysis resources (each node has 1.5 TB RAM, 50TB NVME, and dual Cascade Lake processors). These resources are inaccessible to general users except through the grafana interface due to physical (limited routes and rates) and policy (additional route limitations due to general corporate cyber-security requirements regarded as incompatible with analysis performance) networking barriers. The physical network barriers also prevent direct data analysis using existing HPC cluster compute resources including GPU clusters.

2.3.3.4. Archive

Data gathered on an ASC cluster platform remains relevant for the entire pre-production and production life of the platform, which at Sandia is a span of 5 to 10 years typically. As new questions arise regularly, any data item may suddenly become of critical importance. Thus, retirement of data to tape storage is not part of any desired workflow, and no data is considered "unnecessary" before a cluster's retirement. As cluster retirements tend to occur only after two to three subsequent generations of hardware have happened, performance data from retired clusters is of limited academic or engineering interest; thus, monitoring data from *retired* systems may be archived just to the degree needed to support published results.

2.3.3.5. Share and publish

HPC monitoring data is shared and published within the laboratories using web tools and the corporate-supported solutions for user authentication and access control groups. Currently, monitoring data is shared outside the laboratory for research purposes only after it has been sanitized (specific fields are either anonymized or removed per policy requirements), and data description (if any) is provided in ad hoc formats contained in associated text files or in technical reports. Adoption of standard tools to collect and distribute the data descriptions is needed. Commercially supported tools which support reformatting, slicing, and sanitizing of the data are needed.

The vast majority of the monitoring data is hardware performance counters from unclassified systems which, in the absence of labels associating this data with export-controlled applications, is of no security significance and can be published widely. The volume of this data is petabytes per year, and as such it is:

- Physically impractical to access for academics with their own terabyte class storage servers.
- Financially impractical to access or compute for academics with commercial cloud access.

Collaborating effectively with academics on analysis of such large (PB) data volumes requires co-located and accessible compute and storage resources.

A critical minority of the monitoring data is monitoring data streamed from cluster resource managers and from the applications themselves. This subset of data informs research on alternative scheduling algorithms and on effective use of new and existing hardware architectures for specific code families. This data, in raw form, is filled with labels critical to machine-learning models of system and application behaviors. Collaborating effectively with US citizen academics on analysis of such data without data anonymization can only occur on lab-controlled platforms, as some parts of the data may be export controlled. Reliable, general, commercial tools for anonymization of such data without losing the information key to addressing research issues are not known to be available.

2.3.3.6. Access and reuse

Practical access and reuse of non-sensitive monitoring data outside the labs is limited by infrastructure considerations, as discussed above. Nevertheless, benchmark sets of data for research can be sliced from a large public database if a facility to host it becomes available.

Monitoring data reuse inside Sandia is limited by:

- Access control policies
- Lack of standard access methods to integrate data collected with workflows running on the clusters being monitored.
- Limited means of automated data transfer (CSV format, with no associated descriptive data)
- No standardized means of data discovery (user must explore interactively with grafana)
- Lack of means to identify and share relevant performance data across separate in-house work-groups using the same codes.
- Limited staff available to foster collaboration with potential re-users.

In addition to these barriers, monitoring data reuse in the Tri-lab is limited by:

- Lack of common access control infrastructures
- Lack of means to identify and share relevant performance data across lab borders
- Lack of standard naming of schemas and fields used for particular data sets
- Lack of standard connectivity (network and software) to monitoring data repositories

2.3.3.7. Store and transfer

Current loads:

The raw data acquisition rate is 2-4 PB per year which currently expands to 6 PB/year after indexing. These data flows occur over fast networks (>= 100Gb/s) within the clusters, but analysis results are bottle-necked at 1Gb/s flowing to user's desktops over corporate networks from the grafana server. A daily center-wide full-fidelity data set, such as needed for research on shared storage and multi-cluster scheduling, is 17 TB. To support a 5-year analysis window (historically, an underestimate of Sandia cluster life-span), 30 PB of fast and parallel access storage is needed at current acquisition rates.

Future expectations:

1. Raw monitoring data acquisition will grow 10-100x with the advent of CTS-2 and ATS systems due to:
   - The need to share nodes (at least among jobs of the same user and task type) to achieve efficient hardware and power utilization. Monitoring shared nodes requires monitoring individual processes.

    - The recently created "always-on" capabilities to capture and analyze individual process performance data; current data capture is a few hundred metrics per second per node, as is per-process data. The number of processes per node is of order 100 and planned to go higher.
    - Growth in the number of developers and codes publishing application-specific data to the monitoring system (such as via the Kokkos-LDMS interface in the EMPIRE code [43]).

2. High speed network user-level access (connectivity, routing and APIs) to the monitoring data system from the monitored clusters will be needed to support:
    - Scaling the computational work of longitudinal analyses across current and future volumes of data to run on the monitored clusters themselves rather than solely on the CPUs of the monitoring cluster's fast storage system.
    - Enabling applications to tune themselves at run-time using performance feedback from the monitoring system.

3. Inter-institutional sharing of monitoring data that characterizes common codes and shared workloads will require fast NSE-wide connectivity and common access control infrastructure.

4. Monitoring and analysis storage will be integrated with other fast HPC storage resources to eliminate storage administration burdens from the monitoring team.

5. Monitoring and analysis storage capacity will grow to a capacity of 300-3000 PB, or there will be the equivalent development of time-series databases supporting low-loss compression and fast retrieval combined with lesser physical storage.

6. Data required for relevant HPC performance research will become so large that they are practically only accessible as the combination of both compute resources and data, rather than as raw data files "thrown over the wall" to the public. This will require "store and transfer" to be replaced with "provide well-resourced access to analyze".

2.3.3.8. Gaps, pain points and lessons learned

Table 2-4 lists resource areas and gaps important to the monitoring project. The most important lesson learned is the need to flexibly support access control user expectations and corporate access control requirements.

Table 2-4. Data Infrastructure Gaps for HPC Monitoring

Resource	Current Use	Gaps
10/100Gb network	HPC data collection	Missing 10/100Gb network access to data collected from HPC cluster compute nodes and data-center-hosted workstations (DCW) for ML analyses
1Gb network	Desktop access to Grafana vis server, DCW/cluster access to GPFS, DCW/Cluster access to NFS (home, projects)	Isolated authentication domains (production clusters, monitoring servers, file transfer points) complicate or prohibit data transfers
Database service	'hot' cache for monitoring data	Annual data collection of order 4PB limits efficient (hot) analyses to the most recent week time window, shorter than some HPC jobs.
GPU-based libraries	None	Bulk time-series ML analyses using GPUs are impractical without more GPUs and fast network connectivity between them and the monitoring data.
GPFS bulk storage	Warm store for monitoring data	(1) Current 8PB server will be full in 1-2 years, restricting longitudinal analyses. (2) Inaccessible from HPC cluster nodes for analyses.
NFS	Export destination for query result sets	Capacity & Bandwidth
Lustre	None	Best long-term storage option for HPC analysis of monitoring data is not used due to lack of: 100Gb connectivity, Capacity, Access from DCW.
Co-located compute and data access	Production monitoring & Algorithm R&D	HPC monitoring data of research interest is so big that it cannot be processed at all on student research desktops and it cannot be processed affordably in the cloud. Insufficient storage and network to supply foreign national researchers data access on open lab systems.
Data interoperability framework	None	No staff available to develop custom solutions. No sustainable, corporate-supported solution available in HPC Linux and Linux DCW environments for petabyte data interoperability.
Application-specific job descriptions	Workload characterization, Detecting degraded operations, Detecting performance regressions in CI/CD jobs, Job-sizing estimates.	Ensemble job creators do not publish group job descriptions and individual job descriptions to the monitoring system at submission time. Most applications do not publish run-time job descriptions to the monitoring system.
Authentication & authorization	Access to grafana services and files	Corporate infrastructure implementations difficult to track. Container support needed.

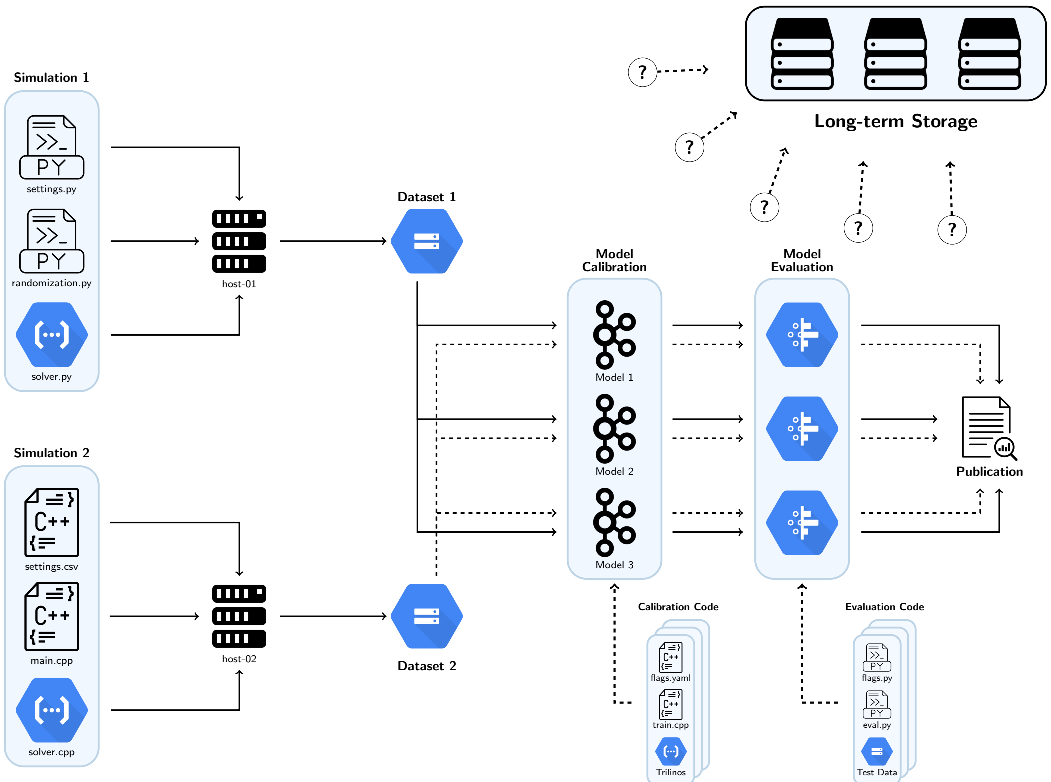


Figure 2-19. Manually ingested forest data is used to initialize ensembles of high fidelity simulations for wildfire ignition and propagation. Simulation results are used to generate fast surrogate neural network-based fire models

2.3.4. Wildfire Digital Twin

Sandia's wildfire digital twin project applies proper orthogonal decomposition (POD) based neural network (NN) techniques to create fast, low-fidelity surrogate models from data generated using high-fidelity simulations with WRF-SFIRE [44], in the service of outer-loop problems like fire control. The inputs and outputs of this NN are projected into reduced-dimensional POD space. The prototype dataflows are illustrated in Figure 2-19.

The HPC model prototype predicts the fire spread in a 1 mi x 1 mi area over a 6-hour time horizon, with snapshots saved every 10 seconds. When scaling area and time horizon to production use, the 2022 Calf Canyon/Hermits Peak (CCHP) fire in New Mexico which burned a total area of 534 mi^2 over 4.5 months provides a useful reference.

2.3.4.1. Plan and design

Researchers created a file-system based design to capture both model context information and bulk data; conventions for directory and file naming must be followed. Files named ".meta"

Table 2-5. Data Software Stack for Wildfire Digital Twin

Analysis & visualization tools	Custom gui, ParaView, Fortran, C/C++, OpenMP, TensorFlow, PyTorch, CUDA, CUDNN
Data interface tools/middleware	Custom python scripts, manual transfers, NetCDF
Data services	File hierarchy conventions indexed with an SQL database
Data architecture & interoperability frameworks	None
Storage, networking, and computing systems	HPC clusters, desktops, on-premises cloud
Org. support, culture, & workforce	Lab and university researchers develop analysis methods and import public data sources. Long-term data curation and sharing needed to convert project to a permanent capability are not supported.

contain information describing and linking input and output data files, as well as high- and low-fidelity hyperparameters.

2.3.4.2. Acquire

Geographic forestry information and historical weather data from U.S. Geological Survey repositories is imported. These datasets are static, relatively coarse (30m resolution), and reasonably small (approximately 10KB/mi^2). Static data are inputs to the high-fidelity WRF-SFIRE code, which generates the fire-spread data used to train the neural network. The output of one WRF-SFIRE run, in our example, is approximately 1.7GB. The precise static data and physical parameters that generate a particular fire trajectory is cataloged to avoid needing to re-run WRF-SFIRE repeatedly. 25 independent WRF-SFIRE runs to generate sufficient data for NN training (15 training, 5 for validation, 5 testing). Using 100 proper orthogonal decomposition (POD) modes yields relative projection error on the order of 1%.

Scaling this process to a realistic CCHP-scale fire yields a modestly increased input data volume of 5MB but an output increase on the order of 500TB (1.7GB / (1 mi^2 x 6 h) x 534 mi^2 x 20 weeks x 168 hrs / 1 week). Neural network size and training times will also substantially increase. The computational complexity scales quadratically in the number of POD modes necessary to capture the new dynamics, which are both large-scale and highly nonlinear. At least an order of magnitude more POD modes will likely be required for sufficient accuracy over such a large spatial domain.

2.3.4.3. Process and analyze

After obtaining 25 fire trajectories, a neural network with 4 hidden layers and 100 neurons per layer is trained to approximate the evolution of the system from time t to $t + \Delta t$. The total training time is approximately 1 core-hour. The resulting NN model can be applied to predict dynamic burn behavior with given initial fire and weather conditions. This low-fidelity model must be referenced from the associated ".meta" file. To predict the evolution of real-world fires, the static fuel and topography data would be on the order of 5MB, and the increase in spatial and time domains would result in output files on the order of 500TB for an event on the scale of the CCHP fire. This would also dramatically increase the NN size and training times. The NN complexity scales quadratically in the number of POD modes necessary to capture the new dynamics, which are both large-scale and highly nonlinear. At least an order of magnitude more POD modes will likely be required for sufficient accuracy over such a large spatial domain.

2.3.4.4. Archive

Currently the project has no archived data due to the small size of the test problem.

2.3.4.5. Share and publish

Forest fires are a public and even international concern. External researchers seeking to validate and improve models of public consequence will need access to the forest data, derived HPC simulation data, and derived NN models. To enable this, the large data and metadata associated with the workflows must all be made publicly available in formats that follow appropriate industry standards enabling automated data translation and search. This in the fullness of time could easily require a public store on the scale of petabytes, accomodating data on the scale of the continental United States forests. Providing an API to the HPC fire data compliant with appropriate public standards will be necessary.

2.3.4.6. Access and reuse

Workflows for access and reuse fall into three broad categories:

- Production/field use of the NN models for wildfire response.
- Periodic reuse of the model generation workflow to ensure all the infrastructure (python and its successors, HPC cluster successors, new WRF-SFIRE or its replacements, file systems and their successors, file formats and their successors, data sources and their import processes, . . .) remain available and capable.
- Sustained, episodic reuse of the model generation workflow to perform R&D needed to support new input types and desired new field capabilities.

Currently researchers believe that the time-scale of wildfire responses and their relative rarity would allow for large model data to be stored on tape and recalled with a latency of 24 hours without undue impact on the wildfire response. Repetitive tape recall, however, will be inefficient for periodic (weekly) workflow testing, so a permanently available, fast-access, comprehensive set of data for workflow validation and continued research is necessary.

2.3.4.7. Store and transfer

The current model prototype is sized (1 mi^2) for the simulation data to be manageable with desktop hardware and 1Gb networks after generation using an HPC cluster.

In the future, we anticipate a collection of ready-to-use fire models and their supporting data spanning major forests in the continental U.S. (based on the prototyped modeling techniques) would consume storage capacity in the realm of 10s of PB. Transferring one fire (500 TB) over a 100Gb link will consume approximately 14 hours; this data rate and storage volume is manageable in a laboratory HPC setting, but prohibitive in most current academic research settings.

2.3.4.8. Gaps, pain points and lessons learned

Table 2-6 lists resource areas and gaps important to the wildfire digital twin project.

Table 2-6. Data Infrastructure Gaps for HPC Monitoring

Resource	Current Use	Gaps
10/100Gb network	In HPC simulation runs	Fast access to bulk data
1Gb network	Desktop/NFS/GPFS data access	None
Database service	Indexing available simulations	Not corporate supported
GPU-based libraries	None	HPC code used is not yet GPU enabled. Real time or faster simulation could be needed at scale.
NFS	Training data storage and low-fidelity model building	Capacity. Read/write rate performance. NFS file permissions/ownership inheritances are unexpected.
Lustre	Training data generation	Permanency
Tape archive	Off-load training data	Rapid discovery and recall of large training data. Easy-to-use automated mirror-to/update-mirror/mirror-from tape utility.
Visualization	Desktop analysis	Remote visualization of cluster stored data.
Data interoperability framework	None	No sustainable, corporate-supported solution available in HPC Linux and Linux DCW environments for terabyte data interoperability.

3. DATA INFRASTRUCTURE GAPS

This chapter responds to completion criterion 2 by identifying gaps in current data infrastructure based on the example applications and workflows outlined in Chapter 2. Plans grounded in these gaps and requirements are proposed in Chapter 5.

The gaps listed here are not unique to the needs of an ASC AML program, but are also relevant to a wide range of emerging data-driven science and engineering activities at the laboratories.

3.1. Current Status and Gaps

This section uses the data technology stack elements outlined in Section 1.1.1 to highlight gaps in existing tri-lab data infrastructures. Where possible, gaps are linked to the example applications and workflows in Chapter 2. Note that gaps are cumulative, that is the gaps at the bottom of the technology stack typically are also gaps at the higher levels of the stack. For example, systems issues, such as network bandwidth and storage limitations limit the high level machine learning tool usage.

3.1.1. Gaps for Analysis and Visualization Tools

This element of the data stack encompasses analysis and visualization tools including AI/ML tools used for AML workflows as well as other science and engineering applications. In many current research workflows, analysis and visualization are treated as a post-processing step: i.e., the simulation completes with data dumped to files in a format supported by a separate visualization tool. This approach may not be sustainable for future AML and data-forward research and engineering workflows where large amounts of data from various sources may need to be analyzed and compared in depth. Some existing machine learning workflows (e.g. PIML) are already moving in new directions, with ML algorithms and simulation libraries integrated into a single application due to the complexity of the data. There are also emerging cases of HPC system monitoring and feedback control with ML, and of simulations where in-situ analyses are used for output reductions and ML model creation at run-time.

1. All data science, analysis, ML and visualization software is not currently design to work efficiently on our ASC HPC resources. Gaps includes the ability of this software to make efficient use of 1) specialized computational resources such as ARM processors and GPUs 2) the full HPC storage heirarchy from tape to burst buffer to cache , 3) distributed memory computational resources to scale to handle large data. An additional concern is the timely, regularly updated installation and testing of machine learning software on our ASC HPC resources. (Cases PIML, HPC Monitoring, Wildfire DT, . . .)

2. There is a need for advanced analysis and visualization tools to support comparison between data sets and drilling down into data using metadata categories, with support for workflow automation. (Cases Radiograph, . . .)

3.1.2. Gaps for Data Interfaces

Data interfaces provide an organizational structure for data such as a description of its structure, type and meaning. We include the programming languages and software libraries that provide the underlying framework for data descriptions that enable data analysis, visualization, and AI/ML workflows. Currently data interfaces are often managed on a per application-team (more rarely, on a per user) basis, often via scripts passed around within teams or technical communities. However, this fragmented approach does not scale well and may lead to discrepancies in data engineering and solutions. Future data science and AML projects are likely to extend current trends of increasingly large collaborations that extend across teams and even institutional boundaries. This may require additional efforts to create shared data interface standards.

Currently, each laboratory is developing simulation data frameworks (eg. SDM, . . .) and metadata tracking tools (eg. ModelSea, GUFI. . .), but often without coordination across laboratories; varying degrees of provenance tracking are supported. Translation of data from one tool to another, even when a common file format is used such as HDF5, is often implemented as a point-to-point operation for a single tool pair.

As AI capabilities improve, additional use cases may emerge requiring increasingly sophisticated data interfaces. For example, there is interest in developing facilities for autonomous scientific discovery, integrating AI models with experimental capabilities. This may create a need for data interfaces and workflows that enable rapid machine access to data at a massive scale with minimal human intervention. This causes a gap with respect to security and quality control as throughput will reach a point where a human will not be able to validate every decision from the AI.

1. There is a lack of clearly specified interfaces for each ASC code such that translator constructions are easily, and mostly automatically, carried out and maintained. (Cases Climate, PIML, HPC Monitoring, Wildfire DT, ...)
2. Too many data interface translations and data transfers in a workflow are inefficient (Cases ICECap, Edge, ...)
3. Lack of data interfaces that support management of multidimensional data ensembles. (Cases Radiograph, ...)
4. Lack of tools for ensuring quality control and ensuring the security of data as it moves through increasingly complex and automated workflows (Case Edge).

3.1.3. Gaps in Data Services

Data services are software and hardware used for maintaining, organizing, and moving data. Examples include database management systems, data warehouse services, data transfer services, and metadata support and indexing tools. Most teams create their own custom software data services, that work only in their local HPC evironment, rendering cross team collaboration and portability difficult at best. There are multiple barriers to cross domain or team workflows. Discovering the data is usually done by either repeating the data generation step or asking a colleague where such data can be found. There is no widely used tool for cataloging and discovery of data. Data is trapped in access-control domains, convenient to the data creators, instead of supporting sharing with others in the National Security Enterprise (NSE). Traditional corporate-supported database services, that typically include SQL and non-SQL technology offerings, are available at each laboratory for small scale, in HPC terms, data storage.

Globus [45][1] is an example of the state of the art in Data Services in DOE. Globus offers a unifying authentication, authorization, data and compute access, and workflow orchestration platform deployed over many sites, including DOE laboratories. Using Globus application have link data resources across sites, for example, one application used DOE supercomputers to generate training data for an AI model that permited real-time control of experiments at DOE X-ray sources.

1. Existing database and data warehousing data services do not scale well to the needed access rates and data volumes common in HPC. Custom data services being implemented in the absence of a shared database solution to support HPC and engineering workflows across the Tri-lab lack a common technical foundations to enable interoperability. (Cases PIML, HPC Monitoring, Wildfire DT, ...)

3.1.4. Gaps in Data architecture

Data architecture includes common tools and standards for enabling interoperability of data and data analysis tools, potentially both within and across institutions. These are often associated with institutional data management and governance standards and provide the fundamental capabilities necessary for FAIR data access and other institutional requirements such as security. They include API libraries (i.e., institutional repositories of supported APIs), metadata frameworks, data virtualization services, and security frameworks (e.g., POSIX security). The term architecture reflects the need to design these systems as an environment for support broad institutional needs rather than those of a specific research group.

Data architecture covers common tools and standards that facilitate interoperability within and across institutions. To the extent that these exist, they are currently implemented mainly within specific programs/projects or facilities and are not coordinated at an institutional or cross-institutional level. Data is often not expected to leave the originating team, and if it does the burden is usually on the recipient to reorganize the data in a way that will fit their needs. AML workflows may require data spanning various sources (e.g., experiments, material data bases, geometry databases, GIS database), possibly generated with different versions of tools and in different formats.

Mixing and match data across tools, versions and formats is very cumbersome in the absence of coordinated data architectures. For example, data formats may not agree because the same

[1] Globus: https://www.globus.org

concept has different names within different teams. Another challenge is units of measurement between teams may vary.

There are emerging tools to help with this such at Sina/Kosh at LLNL, DataSea and Simulation Data Management at SNL, and Data Science Infrastructure (DSI) at LANL. None of the tools currently in use fully support FAIR principles through all stages of their supported workflows. To support enterprise-wide activities composed of many interlinked workflows across multiple sites, data architecture solutions may go through some period when bridging the interoperability frameworks themselves is also needed. A common data architecture need not be rigidly defined across all institutions, but key standards and protocols need to be agreed on to enable interoperability.

Data security is another element of data architecture that is impacted by emerging advanced AI/ML or data-intensive science workflows. Models that are trained on combined datasets that may have different access control requirements, or where compilation of data elements has the potential to increase data sensitivity beyond the sensitivity of the original source datasets, are of particular concern. Currently we lack tools to flexibly assess sensitivity levels and enforce access controls over large combined data sets or derived ML models. Incorporating basic security information and controls at the data architecture level will make it possible to build security into all layers of the data technology stack.

1. Many currently implemented interoperability solutions lack the flexibility needed to support the wide and regularly changing variety of data formats coming from experiments and simulations. (Cases ICECap, Climate, PIML, Wildfire DT, ...)
2. Many currently implemented interoperability solutions lack support for the full span of operating systems used in the labs (Windows, Linux in several major versions, macOS). (Cases PIML, . . .)
3. Most ASC applications do not have the detailed, machine-readable metadata and associated translation tools needed for automatically and sustainably mapping inputs or results to other tools in ASC workflows. (Cases ICECap, Climate, PIML, ...)
4. Current metadata standards and infrastructure do not universally support ability to track origins and transformations of data throughout the data lifecycle, potentially including information like data originator/author, original experiment/code/model settings and parameters, parent/child data set relationships, or other metadata necessary for identifying and understanding relevant data sets. (Cases Ensembles, Bueno, ICECap, Edge, PIML, HPC Monitoring, ...)
5. We currently lack tools to flexibly assess sensitivity levels and enforce security and access controls over large combined data sets or derived ML models. (Cases Bueno, Climate, Edge, HPC Monitoring, ...)

3.1.5. Storage, networking, and computing systems (SNCS)

Storage: ASC supports a variety of HPC storage systems such as tape, shared NFS, shared GPFS, shared Lustre, and cluster-specific storage solutions, are available at each tri-lab site. A variety of methods including quotas, periodically wiped campaign storage, and administrative reminders, are used to minimize the amount of hardware needed in provisioning these storage solutions. Slow storage solutions (e.g. NFS) are often used to enable ubiquitous data access and workflows that avoid the need to develop complex file transfer steps. The largest, fastest storage, such as Lustre instances, are accessible from institutional science and engineering networks only vial file transfer nodes. Lustre is also positioned as "scratch use only" and not provisioned for long-term fast access to reference training data sets for AML work. Even Lustre may be slow relative to certain workload data demands, so a variety of storage buffering strategies are in use or in development.

Outside the direct ASC scope, large data sets of great potential value for ASC modeling and simulation (both AML and conventional models) are often kept on cold hard drives or tape, where they are not discoverable except possibly by their creators.

User access is controlled via group membership that may be assigned on need-to-know basis. The Remote Computing Environments (RCE) project is a current effort to harmonize the processes of authentication and access control across the laboratories.

Object storage could become a major factor in the future as it would create team-based quota, transfer the data format and location issue from individuals to lab-wide data specialized teams. This could also help with cross-labs data sharing and transfer. Currently the most successful technology related to inter-site data movement seems to be Globus, discussed in more detail above.

Networking: ASC HPC cluster architectures generally include 1Gb/s access from compute nodes to NFS, GPFS, or remote workstations and 100Gb/s or better access from compute nodes to shared clusters. Traffic exiting the HPC domain is often trunked through a small number of 100Gb uplinks to institutional networks that host workstations. Inter-site "seamless" networking is being developed for open and restricted networks, while it has long existed for classified networks; however, filesystems are not shared across these networks.

Even with good storage and networking, computations requiring data from various sites are hampered by limitations on ability to move data across sites.

Computing Systems Compute hours are typically assigned per user or project and shared via resource managers such as via SLURM, LSF, Flux, and nagging by system administrators. Generally, cluster compute node access is exclusive though this is expected to change as the rising conventional processor core count or single GPU capacity may make it impossible for some applications to fully utilize a node. The coming ATS and CTS-2 systems will provide expanded GPU access, however the GPU hardware and software technology stack continues to evolve rapidly.

1. I/O access delays due to network architecture (Cases Radiograph, ICECap, Climate, Edge, PIML, HPC Monitoring, ...)
    a) Long I/O times from compute nodes, especially to storage resources on 1Gb/s networks, combined with disk quotas or a maximum allowed job time often reduce the amount of data access or output space available to execute a large workload, such as ML training from massive sets of simulation results.

   b) Where data can be explicitly moved ahead of time to HPC fast scratch or buffer storage, workflows become more complicated to execute, track provenance, and maintain.
   c) Long I/O times to large memory multi-GPU single-node compute server resources outside the HPC storage domains impede machine learning workflow efficiency.
2. Data loss for future analysis (including ML) due to storage capacity limitations (Cases Radiograph, ICECap, Edge, PIML, HPC Monitoring, ...)
   a) Later analyses in general and ML in particular cannot be performed on simulation data that is discarded due to intermediate or permanent HPC storage capacity limits or costs. This leads to redundant computation when a new question is asked about the same set of simulations.
   b) Later analyses in general and ML in particular cannot be performed on experimental data that is "lost" because it is "too large" to store or index in corporate environments. Data sitting in hard drives on shelves or on tape is inaccessible.
3. Research slowed or blocked due to the inaccessibility of unclassified data by university partners. (Cases Climate, PIML, Wildfire DT, ...)
   a) Very large data sets for ML or general model development cannot be moved to university-controlled environments due to storage capacity limitations (local equipment or cloud service costs) or export control issues.
   b) Very large data sets for ML or general model development cannot be used by university researchers in the laboratories owning the data due to storage bandwidth or capacity limitations and compute limitations on networks supporting university collaboration.
4. No high-performance, long-term, multi-site, FAIR-supporting repository is available for the exchange and curation of HPC data and part production data within the tri-lab. (Cases ICECap, Climate, PIML, Wildfire DT, ...)
5. Due to the segregation of storage systems, untracked duplicates of large data consume resources across multiple storage systems at the same network security level. (Cases PIML, HPC Monitoring, Wildfire DT, ...)

3.1.6. Organizations, culture, and workforce (OCW)

Computational science research and engineering efforts at the laboratories are increasing in scale and increasingly requiring collaboration and coordination beyond previous team and institutional boundaries, as exemplified by the multi-laboratory exascale computing project (ECP). Large-scale AML and data science efforts are likely to continue and extend this trend toward cross-cutting collaboration. Exascale computing will also lead to great increases in data volumes from simulations. Currently, most data is managed through extensive human intervention to maintain filesystems and move data as needed. This approach will not be sustainable for future high-volume data science and AML workflows. New tools will be required to enable more efficient human data management as well as to allow potential AML tools to interactively guide data discovery or interface directly with large and diverse data sets. In addition, there will be an increasing need for tools, platforms, and standards to support increasingly large and diverse collaborations.

AML and other cross-cutting data workflows will require cultural changes in the way data is managed and controlled. Although data at the national labs belongs to the institution, the current model for sharing data typically relies on data producers to make data available to others on an as-requested basis, which in practice gives data producers considerable control over data. When data is shared, it is often on an ad-hoc basis using non-standard data and metadata formats that may make it difficult for requesters to correctly interpret and use the data. Shifting this model to one where data is stored in institutionally-accessible repositories and users rely on data catalogs and metadata for data discovery will be a major cultural change. Current data policies lack specific guidance for this scenario and for how the needs of data producers will be balanced against institutional needs for making data more widely accessible. There are some efforts at each lab to have local codes output data in similar fashion (eg. DSI, Sina, Sierra, RAMSES) and some teams are already engaging with each over regarding the establishment of conventions on what these outputs mean.

1. There are no institutional requirements that all data be produced and tracked with methods supporting the traceability and FAIR principles suited to a digital enterprise. (Cases ICECap, Climate, PIML, ...)
2. Inter- and intra-site and inter-application teams are not supported to harmonize meta-data and application data definitions. (Cases ICECap, Climate, PIML, ...)
3. Skills suitable to providing distributed-memory versions of the wide spectrum of ML algorithms re-cast and tuned for distributed-memory physics data on HPC platforms are not being mentored or recruited in sufficient numbers. (Cases PIML, HPC Monitoring, Wildfire DT, . . .)
4. Skills suitable to providing exa-scalable NSE-wide and corporate-wide software and platforms for curated, annotated, provenance-tracked data with high-performance access from clusters and high-end servers are not being mentored or recruited in sufficient numbers. (Cases ICECap, PIML, ...)
5. Skills suitable to providing exa-scale file systems based on Lustre, GPFS, or other high-performance technologies that are highly reliable and scalable to NSE-wide shared use are not being mentored or recruited in sufficient numbers. (Cases ICECap, PIML, HPC Monitoring, ...)

3.2. Anticipated data-infrastructure needs for the ASC AI4ND strategy

The NNSA strategy, Artificial Intelligence for Nuclear Deterrence (AI4ND) [2], discusses potential impact of AI/ML in all four phases of the weapon life cycle: Discovery; Design Optimization; Manufacturing and Certification; and Deployment and Surveillance (DDMD). An implicit requirement for the use of AI in each of these phases is a robust data infrastructure. This section will identify data-infrastructure requirements and gaps needed to support each phase of DDMD.

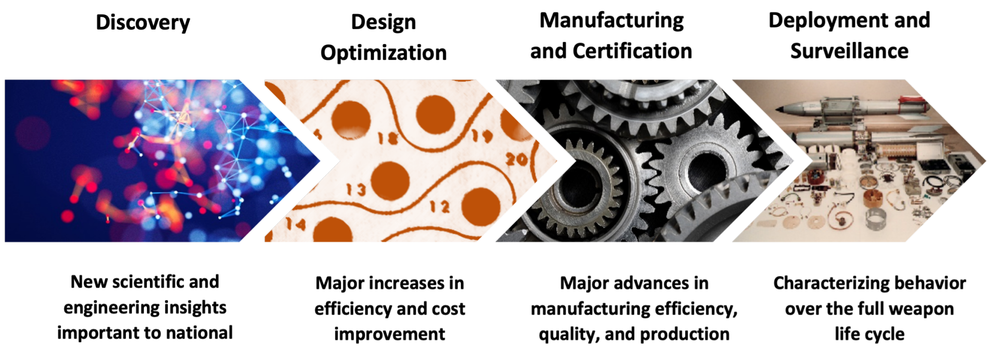


Figure 3-1. Four phases of the DDMD weapon life cycle [3, 2].

3.2.1. Data infrastructure for Discovery

Scientific discovery, particularly in areas such as material science, that meet the stringent needs of our weapon programs is a critical part of our ND program. Machine learning already plays an important role in discovery as demonstrated by the ICECap and PIML examples.

Recent advances in areas like generative AI and foundation models [46, 47] have massive potential to accelerate scientific discovery [3]. The power and potential of foundation models is that they can "develop keen insight and discover meaningful patterns in vast troves of data, that may initially seem uncorrelatable" [3]. One early success is from [48] where they demonstrated transfer learning to calibrate multi-fidelity simulation and experimental data for intertial-confinement fusion.

There are numerous data-infrastructure challenges to support scientific discovery. As demonstrated in 2.2.2 and 2.3.2, our current infrastructure makes it difficult to access and combine multi-fidelity experimental and simulation data. Proper support of new capabilities like foundation models that find, access, and incorporate data from across the complex will require a DOE data ecosystem that implements FAIR practices. Much of the observational, experimental, and simulation data generated for NNSA is managed by teams, not programs, using non-standard interfaces and home-grown tools, and managed in repositories and networks disconnected from communities that could use it.

3.2.2. Data infrastructure for Design Optimization

Machine learning used for design optimization and uncertainty quantification has potential to significantly reduce weapon life cycle time [3, 5] through deployment of ML-surrogates that run 10-100x faster than conventional simulations [49, 50, 51].

Sophisticated workflow tools are necessary to adequately sample design space and robustly handle inevitable code failures. Data management tools are needed to associate metadata for simulations with the results that are stored and used to train surrogates. The data that is used to train specific versions of surrogates, along with the model description (architecture,

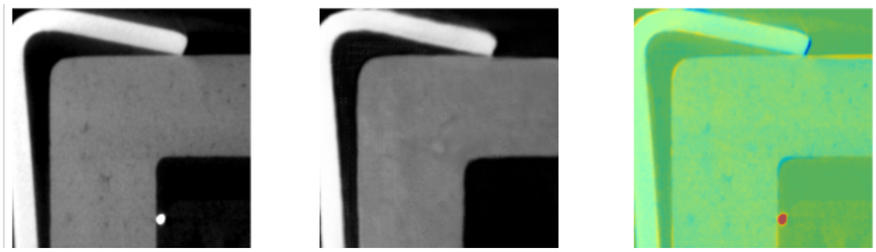


Figure 3-2. Deep learning is being used to detect defects on manufactured parts. Image shows a defects from a current transformer [52].

hyperparameters, etc.), must also be stored for reproducibility of model predictions. Furthermore, although VVUQ for ML is an open area of research, knowing how a model is trained can inform VVUQ for that model when it is being applied to a specific domain.

3.2.3. Data infrastructure for Manufacturing and Certification

Production and certification of weapon parts is one of the most time-consuming tasks in the weapon life cycle. Machine learning has potential accelerate this process through methods that automate testing and inspection, predict performance of manufactured parts, and detect defects caused by various manufacturing processes (as illustrated in Figure 3-2).

The data infrastructure required to fully-integrate ML and automation for manufacturing and certification requires significant advances. Within the production facility, the data-infrastructure should enables *in situ* analysis capabilites on the devices [53] as well as signifcant capacity for data-capture within the facility. To enable rapid co-design between design teams at DOE laboratories and production teams at facilities, we will need a fully connected and performant data ecosystem within the DOE complex.

3.2.4. Data infrastructure for Deployment and Surveillance

As mentioned in the AI4ND strategy, AI/ML will be used to “automatically digest technical documents, compare surveillance data to computational simulation data of its digital twin [54, 55], and create summaries and perform root-cause analysis.” [2].

Data infrastructure to support deployment and surveillance in the way it’s envisioned in the AI4ND strategy document requires significant changes. Close ties between physical systems at production facilities and digital twins (potentially at DOE laboratories) will require high-performance complex-wide data ecosystem. Similarly, the foundational models (i.e., large-language models) necessary to digest, compare, and generate technical documents will require access to a broad set of training data across the complex.

4. DATA-INFRASTRUCTURE REQUIREMENTS TO ENABLE PRODUCTION USE OF AML

This chapter describes a set of high-level data-infrastructure requirements needed to support production use of AML across the National Security Enterprise. The requirements derive from data-infrastructure gaps described in Chapter 3 for existing application workflows and the anticipated needs of the AI4ND strategy.

4.1. Requirement 1: FAIR data principles within NSE

The aspiration of implementing FAIR data-management principles (described in detail in Section 1.2.1) for NSE is quite challenging and will require an unprecedented level of coordination between the individual sites and cultural changes within our organizations. Currently, each site operates as its own entity using different protocols and methods to connect networks, authenticate users and authorize access. Beyond institutional changes needed to support FAIR, research teams do not have the skills, experience, knowledge of tools, or ***desire*** to manage data in compliance with FAIR principles.

These requirements are necessary to achieve FAIR practices within NSE:

(A) Agreement from each of the sites on common methods and security protocols that enable ***automated cross-site access*** to open, sensitive, and classified networks. Data and systems need to be accessible by human users, applications, and workflows.

(B) Complex-wide **adoption of standard practices in data governance and data management**. This may require workforce investments and teaming with data-management professionals to help teams understand how to manage data and use data-management tools, and requirements for data-management plans for ASC-funded projects (now a requirement for NSF [56], NIH [57], and Office of Science [58]).

The benefits of implementing capabilities necessary for FAIR go well beyond just AML. They will enable more effective human collaboration for multi-site teams, consistent processes for access to additional resources besides data (e.g., HPC systems), and data-sharing between design teams, researchers, and production facilities.

4.2. Requirement 2: Performant Data Storage and Networking Technologies and Systems

Storage and networking are the pillars of any data-infrastructure. The ASC program historically emphasizes high-performance storage within our advanced simulation and computing platforms, but an enterprise-wide infrastructure with access to HPC systems, large data repositories, production and experimental facilities will require strategic investments and partnering across programs and institutions far beyond what we have done in the past. Further, the diversity of storage and networks within individual institutions (as described in Section 3.1.5), including HPC storage, network-mounted file systems, cloud storage and tape archives are inadequate for today's workloads.

An execution plan to address storage and networking technologies in support of AI4ND should include the following:

(C) R&D for methods that **integrate and automate management of data across storage hierarchies** within a single institution. This including HPC storage, cloud storage, databases, and storage on remote edge devices and AI-accelerators.

(D) R&D and *infrastructure investments* to **address capacity and performance requirements for emerging AI workflows**. What infrastructure is required to support emerging AI capabilities (e.g., generative AI)? What performance guarantees are needed? How do these requirements guide investments in storage and network infrastructure?

4.3. Requirement 3: Federated Data Environment

The scale and diversity of data from the various DOE laboratories, described in both the examples and anticipated support for DDMD is too large and complex for a single entity to manage. Instead, we will need a federated infrastructure that incorporates storage distributed across the NNSA complex into a common virtual data ecosystem. Due to strict security constraints of NSE networks, solutions to achieve data-federation across NNSA sites is more complicated than similar efforts for open science [59].

Requirements for a federated data-management across NSE include:

(E) NNSA community efforts to agree on methods for **federated resource management**, particularly for storage, network, computing. Workflows that span administrative domains have an expectation that resources are part of a "collective" and can be automatically allocated and scheduled to support the workflow.

(F) R&D into **network-aware algorithms for geographically-distributed workflows** that decide when to move data and when to process data locally.

4.4. Requirement 4: Data Access Interfaces and Tools

Integrating data from diverse pool of programs and sites is a significant challenge. Data includes a mix of design (e.g., CAD) experimental, observational, simulation, documents, databases. The interfaces, formats, and tools used to generate, process, and analyze/visualize data provide specific capabilities for the community they service (e.g., analyst, developer, computational scientist). Mandating specific application-programming interface (API)s, data formats, and tool chains is not achievable or constructive. Rather, we encourage a flexible approach of publishing any information needed to make use of the data. This flexibility would allow, for example, foundation models and large language models to extract knowledge from a diverse set of data.

The requirement reads as follows:

(G) Adoption or development of **methods for publishing** APIs, data format, tools, and other information necessary to make use of the data. This information should be available with or linked to any curated data shared using the FAIR principles described above.

5. EXECUTION PLAN

The goal of this chapter is to propose actionable steps toward achieving "Production use of Machine Learning and Artificial Intelligence" in the NSE based on requirements set in previous chapters.

5.1. Build an ASC Data Infrastructure

We are building an ASC Data Infrastructure that will require developing and fostering partnerships, setting new expectations for project teams, and R&D to address technical gaps.

5.1.1. Build partnerships

- **Develop *committed* institutional partnerships**. Capabilities that enable the type of integrated data access and computing required for production use of AML across the complex will require program leaders and laboratory leaders to implement significant changes to infrastructure at each site. Develop a cost/benefit analysis and devise incentives to encourage the laboratories to collaborate. As an example, it would benefit the complex, including production agencies, to have a complex-wide cybersecurity team to implement consistent processes and methods for access to the complex.
- **Develop program partnerships within NNSA**: Establish defined roles for ASC members in NNSA programs focused on data infrastructure. The complex appears to have many different initiatives like Cerberus, PRIDE, Big Data Pilot, and others. It will be important for ASC to be a key collaborator on these topics.
- **Consider continuing program partnerships within other DOE offices**. Partner with DOE ASCR on a new large program focused on AI for DOE and make sure data-infrastructure is a priority funding item for that program. Any DOE infrastructure program, for example the recent DOE plans for a High Performance Data Facility [60], should ensure the NNSA mission needs are supported.

5.1.2. Define a set of data-management expectations for project teams

- **Implement data-management plans**. Establish guidelines for developing data-management plans for production and research teams. Develop incentives to encourage teams to participate, build an infrastructure to manage and search through plans from other teams.

- **Integrate ASC Data Management Professionals.** Identify, recruit, and integrate data-management professionals, similar to the integration of software-quality engineers, into program sub-elements to help develop plans for managing datasets, curating data, etc.

5.1.3. Perform R&D to address technical gaps

- Perform R&D in data services, storage, analysis and visualization, and other domains that integrate and automate management of data across storage hierarchies and distributed storage (e.g., cloud) within an institution.

- Perform R&D and co-design to effectively leverage data-infrastructure capabilities of hardware, storage, edge devices developed for industry use of AI for workflows relevant to the NNSA mission.

- Perform R&D focused on data-management methods for large-scale ensembles of simulation, experimental, or observational data used for AML, experimental design, optimization, etc.

- Perform R&D for data-federation within a secure environment.

5.2. Key Proposed Milestones

FY25: Deploy a prototype ASC Data Explorer for the NSE (see Section 5.3). Recommended collaboration between CSSE and FOUS.

FY25: Design workflow-benchmarks to evaluate data-management and data-infrastructure performance and capabilities. Focus initially on a single institution's data-infrastructure (e.g., HPC, cloud, database, ...). Recommend collaboration between Defense Applications and Modeling (DAM) and Computational Systems and Software Engieering (CSSE) sub-elements of the ASC program.

FY27: Deploy an NSE-accessible foundation model [46] of mission relevance (e.g., knowledge, design, ...).

FY28: Demonstrate a simple workflow (benchmark) that executes across at least two sites of the NNSA complex.

FY30: Deploy a federated NNSA Data Infrastructure

5.3. A proposed project for a prototype ASC Data Explorer

The DOE Office of Scientific and Technical Information (OSTI) has a site called the DOE Data Explorer[1] with the objective of providing a "search tool for finding DOE-funded, publicly available, scientific data records submitted by data centers, repositories, and other organizations." The data repository provides the "Findable" piece of FAIR data practices discussed in Section 1.2.1. In this section, we propose an NNSA-wide capability similar to the DOE Data Explorer that enables NNSA staff to publish and find sensitive datasets collected and curated by teams in the ND program.

There are several important differences between the DOE Data Explorer and the ASC Data Explorer:

- **Sensitivity of the Data**: All data on the DOE Data Explorer has to be unclassified and "publicly available." The ASC Data Explorer needs to support datasets with a range of classifications and access controls.
- **Authentication & Authorization**: Since the DOE Data Explorer is for publicly available data, the site does not require any authentication or authorization to access the database, and many of sites that manage the data do not require authentication to access the data. An ASC Data Explorer would need to use both authentication and authorization to resolve identity and access permissions of the user.
- **Access to Data**: The DOE Data Explorer only manages the metadata about datasets. It does not host the data. The sites that host the data, however, make it accessible through the internet. There are no special restrictions other than perhaps authentication, required. Like the DOE Data Explorer, we expect the ASC Data Explorer to require individual sites to manage their own data, however, each site has it's own authentication and authorization methods, making accessibility of the data a significant challenge.

5.3.1. Implementation of an ASC Data Explorer

Recent progress of the ASC Sustainable Scientific Software (S^3) community and the tri-lab Remote-Computing Enablement (RCE) is encouraging. A tri-lab accessible database of ASC-generated and curated datasets could complement those efforts nicely and could build on some of thier efforts, particularly around authentication.

The minimum requirements for an ASC Data Explorer are:

- **Repository for metadata**: Just like the OSTI site for the DOE Data Explorer, ASC needs a single repository for metadata at the appropriate networks. This site should be equally accessible to all sites in the complex.

[1] DOE Data Explorer: https://www.osti.gov/dataexplorer

- **Authentication**: Independent of host of ASC Data Explorer, the mechanism used to authenticate the user should be independent of the hosting site. For sites on a restricted network, the most reasonable authentication could be DOE OneID, badge authentication for HSPD-12 badge holders.
- **Authorization**: Like authentication, the tri-labs and production facilities need to agree on a way to set and enforce access controls.
- **Metadata Descriptions**: The repository needs to support flexible and extensible metadata. See Section 1.2.2 for a more detailed discussion of metadata.
- APIs and Tools: The repository needs to support an API that enables each site to search, register, and retrieve metadata about dataset.

As discussed in Section 1.1.2, the most important (and perhaps most difficult) challenge of standing up a dataset repository is providing incentives for using this repository. The first phase of an ASC Data Explorer should provide a way to easily capture and expose curated datasets being generated by and for the ASC program; however, putting the burden of sharing data on the project PIs might be overwhelming. Employing dedicated data-management professionals in the early phases might be the most effective way to seed the effort. Putting rewards on external usage of the data could encourage teams to share more freely.

6. CONCLUSIONS

A robust data-infrastructure is a key enabler for a National Security Enterprise capability for advanced machine learning (AML). This report describes results of a tri-lab milestone to assess the state of our current data infrastructure, identify data-infrastructure requirements for anticipated advances in AML, and propose an execution plan to address gaps and requirements.

To assess the data-infrastructure gaps, teams at LANL, LLNL, and SNL evaluated nine different representative applications, described in detail in Chapter 2. Our milestone team interviewed the application teams to identify technology gaps within the data-technology stack elements outlined in Section 1.1.1. Chapter 3 presents a detailed summary of these gaps.

Chapter 4 presents a set of data-infrastructure requirements to enable production use of AML across the NSE. These include the adoption of FAIR data practices across the complex, high-performance storage and networking technologies, a federated data environment, and data-management that enables publication of data-access interfaces.

The final chapter, Chapter 5, is a proposed "execution plan" toward a rather ambitious multi-site NSE data infrastructure. The actions outlined in the execution plan include significant improvements to institutional and program collaboration and coordination, setting consistent expectations for data-management within our project teams, and strategic R&D to address current and anticipated technical gaps.

REFERENCES

[1] T. Hoang, R. Neely, J. Pruet, and E. Strack, "Asc fy23 implementation plan revision 0," National Nuclear Security Administration, Tech. Rep., 2022. [Online]. Available: https://www.osti.gov/servlets/purl/1889541

[2] National Nuclear Security Administration, "Artificial Intelligence for Nuclear Deterrence Strategy 2023," To Be Released, 2023.

[3] R. Stevens, J. Carter, J. Feddema, D. Kothe, R. Neely, and J. Pruet, "Advanced Research Directions on AI for Science, Energy, and Security: Report on Summer 2022 Workshops," Department of Energy, Tech. Rep., March 2023. [Online]. Available: https://www.anl.gov/ai-for-science-report

[4] National Nuclear Security Administration, "Advanced Simulation and Computing: Computing Strategy," 2022. [Online]. Available: https://www.energy.gov/sites/default/files/2023-03/2022%20ASC%20Computing%20Strategy.pdf

[5] ——, "Advanced Simulation and Computing: Simulation Strategy," 2022. [Online]. Available: https://www.energy.gov/sites/default/files/2023-03/2022%20ASC%20Simulation%20Strategy.pdf

[6] Select Committee on Artificial Intelligence of the National Science and Technology Council, "National Artificial Intelligence Research and Development Strategic Plan 2023 Updates," United States Office of Science and Technology Policy, Tech. Rep., May 2023. [Online]. Available: https://www.whitehouse.gov/wp-content/uploads/2023/05/National-Artificial-Intelligence-Research-and-Development-Strategic-Plan-2023-Update.pdf

[7] National Academies of Sciences, Engineering, and Medicine, *Charting a Path in a Shifting Technical and Geopolitical Landscape: Post-Exascale Computing for the National Nuclear Security Administration (2023)*. The National Academies Press, 2023. [Online]. Available: https://doi.org/10.17226/26916

[8] D. International, *DAMA-DMBOK: Data Management Body of Knowledge (2nd Edition)*. Denville, NJ, USA: Technics Publications, LLC, 2017. [Online]. Available: https://technicspub.com/dmbok2/

[9] DoD Responsible AI Working Council, "U.S. Department of Defense Responsible Artificial Intelligence Strategy and Implementation Pathway," Department of Defense, Tech. Rep., May 2021. [Online]. Available: https://www.ai.mil/docs/RAI_Strategy_and_Implementation_Pathway_6-21-22.pdf

[10] S. Doebling, V. Feagin, M. Ham, and J. Ahrens, "Data infrastructure to support the application of artificial intelligence and machine learning in the NA-11 portfolio at LANL," Weapons Physics Directorate (ALDX). Los Alamos National Laboratory, Los Alamos, NM (United States), Tech. Rep., 2022, LA-UR-22-30302.

[11] M. D. W. et al, "The FAIR guiding principles for scientific data management and stewardship," *Scientific Data*, vol. 3, no. 1, p. 160018, 2016. [Online]. Available: https://doi.org/10.1038/sdata.2016.18

[12] ISO/IEC, "Information technology — Metadata registries (MDR) — Part 1: Framework," International Organization for Standardization, Standard, Jan. 2023.

[13] NISO (2004), "Understanding metadata," National Institute of Standards Organization, Bethesda, MD, Tech. Rep., 2004. [Online]. Available: http://www.niso.org/publications/press/UnderstandingMetadata.pdf

[14] D. Costanzo, J. Cranshaw, S. Gadomski, S. Jezequel, A. Klimentov, G. Miotto, D. Malon, G. Mornacchi, P. Nemethy, T. Pauly, H. Von der Schmitt, and D. Barberis, "Atlas metadata task force," *Physical Review Letters - PHYS REV LETT*, 01 2007.

[15] M. Baca, Ed., *Introduction to metadata*, 3rd ed. Getty Publications, 2016. [Online]. Available: https://www.getty.edu/publications/intrometadata/

[16] A. Ball, *Review of Data Management Lifecycle Models*. UK United Kingdom: University of Bath, Feb. 2012.

[17] DOE Office of Science, "Statement on digital data management," https://science.osti.gov/Funding-Opportunities/Digital-Data-Management, Jan. 2022.

[18] Advanced Scientific Computing Advisory Committee, "Report on the current policies and practicies for disseminating research results in fields relevant to the advanced scientific computing research program," Department of Energy, Tech. Rep., Jun. 2011. [Online]. Available: https://science.osti.gov/-/media/ascr/ascac/pdf/reports/Data_Dissemination_Final_Report.pdf

[19] Basic Energy Sciences Advisory Committee, "BESAC report on public access to research results," Department of Energy, Tech. Rep., Aug. 2011. [Online]. Available: https://science.osti.gov/-/media/bes/besac/pdf/Public_Access_to_Data_Transmittal_letter_and_Report.pdf

[20] Biological and Environmental Research Advisory Committee, "Current polices and practices for disseminating research results in the fields relevant to the biological and environmental research program," Department of Energy, Tech. Rep., Jun. 2011. [Online]. Available: https://science.osti.gov/-/media/ber/berac/pdf/BER_data_report.pdf

[21] Fusion Energy Sciences Advisory Committee, "FESAC panel report addressing policies and practices influencing the dissemination of research results," Department of Energy, Tech. Rep., Jun. 2011. [Online]. Available: https://science.osti.gov/-/media/fes/pdf/workshop-reports/FESAC_research_dissemination_subcommittee_report.pdf

[22] High Energy Physics Advisory Panel, "Report of the HEPAP sub-committee on the dissemination of research results," Department of Energy, Tech. Rep., Jun. 2011. [Online]. Available: https://science.osti.gov/-/media/hep/hepap/pdf/Dissemination_Report.pdf

[23] Nuclear Science Advisory Council, "Report of the nsac sub-committee on public access to research results," Department of Energy, Tech. Rep., Jun. 2011. [Online]. Available: https://science.osti.gov/-/media/np/nsac/pdf/docs/NSAC_PARR_report_final.pdf

[24] J. D. Hutchinson, J. L. Alwin, A. R. Clark, T. E. Cutler, M. J. Grosskopf, W. Haeck, M. W. Herman, N. A. Kleedtke, J. R. Lamproe, R. C. Little, I. J. Michaud, D. Neudecker, M. E. Rising, T. A. Smith, N. W. Thompson, and S. A. Vander Wiel, "Euclid: A new approach to improve nuclear data coupling optimized experiments with validation using machine learning [slides]," *OSTI*, Jul. 2022. [Online]. Available: https://www.osti.gov/biblio/1898108

[25] A. Biswas, D. C. Francom, D. J. Luscher, J. N. Plohr, D. J. Walters, and S. K. Sjue, "The Impala's horn applied to posterior samples of Ti-6Al-4V strength model parameters," Los Alamos National Laboratory, Los Alamos, NM (United States), Tech. Rep., 2021, LA-UR-21-22189.

[26] S. Sjue, J. Ahrens, A. Biswas, D. Francom, E. Lawrence, D. Luscher, and D. Walters, "Fast strength model characterization using Bayesian statistics," *AIP Conference Proceedings*, vol. 2272, no. 1, p. 070043, 11 2020. [Online]. Available: https://doi.org/10.1063/12.0000882

[27] J. S. Smith, B. Nebgen, N. Mathew, J. Chen, N. Lubbers, L. Burakovsky, S. Tretiak, H. A. Nam, T. Germann, S. Fensin, and K. Barros, "Automated discovery of a robust interatomic potential for aluminum," *Nature Communications*, vol. 12, no. 1, February 2021. [Online]. Available: https://doi.org/10.1038%2Fs41467-021-21376-0

[28] E. Hertel, R. Bell, M. Elrick, A. Farnsworth, G. Kerley, J. McGlaun, S. Petney, S. Silling, P. Taylor, and L. Yarrington, "CTH: A software family for multi-dimensional shock physics analysis," in *Shock Waves@ Marseille I: Hypersonics, Shock Tube & Shock Tunnel Flow*. Springer, 1995, pp. 377–382.

[29] C. Sweeney, O. Korobkin, M. Reid, J. Ahrens, and S. De, "Metadata management in support of training data selection for scientific machine learning workflows," 2023, Poster presented at Conference on Data Analysis (CoDA) 2023. Santa Fe, NM, USA. Abstract: LA-UR-23-20816, Poster: LA-UR-23-21933.

[30] M. Hossain, B. T. Nadiga, O. Korobkin, M. L. Klasky, J. L. Schei, J. W. Burby, M. T. McCann, T. Wilcox, S. De, and C. A. Bouman, "High-precision inversion of dynamic radiography using hydrodynamic features," *Opt. Express*, vol. 30, no. 9, pp. 14 432–14 452, Apr 2022. [Online]. Available: https://opg.optica.org/oe/abstract.cfm?URI=oe-30-9-14432

[31] Z. Huang, M. Klasky, T. Wilcox, and S. Ravishankar, "Physics-driven learning of wasserstein gan for density reconstruction in dynamic tomography," *Appl. Opt.*, vol. 61, no. 10, pp. 2805–2817, Apr 2022. [Online]. Available: https://opg.optica.org/ao/abstract.cfm?URI=ao-61-10-2805

[32] J. Ahrens, S. Jourdain, P. O'Leary, J. Patchett, D. H. Rogers, and M. Petersen, "An image-based approach to extreme scale in situ visualization and analysis," in *SC '14: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 2014, pp. 424–434.

[33] D. Boehme, T. Gamblin, D. Beckingsale, P.-T. Bremer, A. Gimenez, M. LeGendre, O. Pearce, and M. Schulz, "Caliper: Performance introspection for hpc software stacks," in *SC '16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 2016, pp. 550–560.

[34] S. K. Gutierrez and H. P. Pritchard, "Toward Well-Provenanced Computer System Benchmarking: An Update," Los Alamos National Laboratory, Los Alamos, NM (United States), Tech. Rep., 2021, LA-UR-21-29427.

[35] Splunk.com, "Splunk," 2023, [Online; accessed 08-August-2023]. [Online]. Available: http://www.splunk.com

[36] J. L. Peterson, B. Bay, J. Koning, P. Robinson, J. Semler, J. White, R. Anirudh, K. Athey, P.-T. Bremer, F. Di Natale, D. Fox, J. A. Gaffney, S. A. Jacobs, B. Kailkhura, B. Kustowski, S. Langer, B. Spears, J. Thiagarajan, B. Van Essen, and J.-S. Yeom, "Enabling machine learning-ready hpc ensembles with merlin," *Future Generation Computer Systems*, vol. 131, pp. 255–268, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X22000322

[37] M. Michael Glass, M. Heinstein, H. Padilla, S. Roberts, and A. Fate, "ASC integration with DEE and the ADE Pathfinder's Progress," Sandia National Laboratories(SNL-NM), Albuquerque, NM (United States), Tech. Rep. SAND2023-03028PE, 5 2023.

[38] R. Oldfield, S. L. B. Kramer, A. Rushdi, E. Acquesta, J. Emery, P. Kuberry, J. Ray, S. Ackerman, E. Cyr, G. Saavedra, C. Hughes, S. Cardwell, and J. Smith, "The asc advanced machine learning initiative at sandia national laboratories: Fy21 accomplishments and fy22 plans." *OSTI*, 10 2021. [Online]. Available: https://www.osti.gov/biblio/1895016

[39] A. Frankel, C. M. Hamel, D. Bolintineanu, K. Long, and S. Kramer, "Machine learning constitutive models of elastomeric foams," *Computer Methods in Applied Mechanics and Engineering*, vol. 391, p. 114492, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045782521007027

[40] W. P. Collins, K. M. Gutierrez, S. R. McRee, D. N. Sands, and A. C. Yaklin, "Friendly extensible transfer tool beta version, version 00," 4 2016. [Online]. Available: https://www.osti.gov/biblio/1326632

[41] B. Schwaller, N. Tucker, T. Tucker, B. Allan, and J. Brandt, "Hpc system data pipeline to enable meaningful insights through analysis-driven visualizations," in *2020 IEEE International Conference on Cluster Computing (CLUSTER)*, 2020, pp. 433–441.

[42] A. Agelastos, B. Allan, J. Brandt, A. Gentile, S. Lefantzi, S. Monk, J. Ogden, M. Rajan, and J. Stevenson, "Continuous whole-system monitoring toward rapid understanding of production hpc applications and systems," *Parallel Computing*, 5 2016. [Online]. Available: https://www.osti.gov/biblio/1263594

[43] O. Aaziz, B. Allan, J. Brandt, J. Cook, K. Devine, J. Elliott, A. Gentile, S. Hammond, B. Kelley, L. Lopatina, S. Moore, S. Olivier, K. Pedretti, D. Poliakoff, R. Pawlowski, P. Regier, M. Schmitz, B. Schwaller, V. Surjadidjaja, M. S. Swan, N. Tucker, T. Tucker, C. Vaughan, and S. Walton, "Integrated system and application continuous performance monitoring and analysis capability," *OSTI*, 9 2021. [Online]. Available: https://www.osti.gov/biblio/1819812

[44] J. Mandel, J. D. Beezley, and A. K. Kochanski, "Coupled atmosphere-wildland fire modeling with wrf-fire version 3.3," *Geoscientific Model Development Discussions*, vol. 4, no. 1, pp. 497–545, 2011.

[45] I. Foster and C. Kesselman, "Globus: A metacomputing infrastructure toolkit," *The International Journal of Supercomputer Applications and High Performance Computing*, vol. 11, no. 2, pp. 115–128, 1997.

[46] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill *et al.*, "On the opportunities and risks of foundation models," *arXiv preprint arXiv:2108.07258*, 2021. [Online]. Available: https://doi.org/10.48550/arXiv.2108.07258

[47] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving language understanding by generative pre-training," OpenAI, 2018, https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

[48] K. D. Humbird, J. L. Peterson, B. K. Spears, and R. G. McClarren, "Transfer learning to model inertial confinement fusion experiments," *IEEE Transactions on Plasma Science*, vol. 48, no. 1, pp. 61–70, 2020.

[49] R. K. Tripathy and I. Bilionis, "Deep uq: Learning deep neural network surrogate models for high dimensional uncertainty quantification," *Journal of computational physics*, vol. 375, pp. 565–588, 2018.

[50] E. Newman, L. Ruthotto, J. Hart, and B. van Bloemen Waanders, "Train like a (var) pro: Efficient training of neural networks with variable projection," *SIAM Journal on Mathematics of Data Science*, vol. 3, no. 4, pp. 1041–1066, 2021.

[51] G. Kluth, K. D. Humbird, B. K. Spears, J. L. Peterson, H. A. Scott, M. V. Patel, J. Koning, M. Marinak, L. Divol, and C. V. Young, "Deep learning for NLTE spectral opacities," *Physics of Plasmas*, vol. 27, no. 5, p. 052707, 05 2020. [Online]. Available: https://doi.org/10.1063/5.0006784

[52] E. A. Donahue, T.-T. Quach, K. Potter, C. Martinez, M. Smith, and C. D. Turner, "Deep learning for automated defect detection in high-reliability electronic parts," in *Applications of Machine Learning*, vol. 11139. SPIE, 2019, pp. 30–40.

[53] L. Scime, J. Haley, W. Halsey, A. Singh, M. Sprayberry, A. Ziabari, and V. Paquit, "Report on progress of correlation of in-situ and ex-situ data and the use of artificial intelligence to predict defects," Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States), Tech. Rep., 2020.

[54] Y. He, J. Guo, and X. Zheng, “From surveillance to digital twin: Challenges and recent advances of signal processing for industrial internet of things,” *IEEE Signal Processing Magazine*, vol. 35, no. 5, pp. 120–129, 2018.

[55] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, “Characterising the digital twin: A systematic literature review,” *CIRP journal of manufacturing science and technology*, vol. 29, pp. 36–52, 2020.

[56] National Science Foundation, “Preparing your data management plan,” Jan. 2011. [Online]. Available: https://new.nsf.gov/funding/data-management-plan

[57] National Institutes of Health (NIH), “Final NIH policy for data management and sharing,” Oct. 2020, Notice Number: NOT-OD-21-013. [Online]. Available: https://grants.nih.gov/grants/guide/notice-files/NOT-OD-21-013.html

[58] National Science Foundation, “Hi,” Jul. 2014, part of the DOE Public Access Plan. [Online]. Available: https://new.nsf.gov/funding/data-management-plan

[59] A. Zarochentsev, A. Kiryanov, A. Klimentov, D. Krasnopevtsev, and P. Hristov, “Federated data storage and management infrastructure,” in *Journal of Physics: Conference Series*, vol. 762, no. 1. IOP Publishing, oct 2016, p. 012016. [Online]. Available: https://dx.doi.org/10.1088/1742-6596/762/1/012016

[60] HPCWire staff writer. (2023, Mar.) DOE requests proposals from national labs to lead the high performance data facility. [Online]. Available: https://www.hpcwire.com/off-the-wire/doe-requests-proposals-from-national-labs-to-lead-the-high-performance-data-facility/

DISTRIBUTION

Email—Internal

Name	Org.	Sandia Email Address
Rob Hoekstra	1420	rjhoeks@sandia.gov
John Feddema	1460	jtfedde@sandia.gov
James Stewart	1440	jrstewa@sandia.gov
Tom Klitsner	9320	tklitsn@sandia.gov
Technical Library	1911	sanddocs@sandia.gov

Email—External

Name	Company Email Address	Company Name
Tim Randles	trandles@lanl.gov	LANL
Becky Springmeyer	springmeyer1@llnl.gov	LANL
Thuc Hoang	thuc.hoang@nnsa.doe.gov	NNSA
David Etim	david.etim@nnsa.doe.gov	NNSA
Simon Hammond	simon.hammond@nnsa.doe.gov	NNSA

Sandia National Laboratories