ASCR Workshop on In Situ Data Management

Bethesda North Marriott Hotel and Conference Center
5701 Marinelli Road, Rockville, MD 20852
January 28 - 29, 2019
https://www.orau.gov/insitudata2019/

Pre-Workshop Document

1. Introduction

The Department of Energy (DOE) Office of Advanced Scientific Computing Research (ASCR) will convene a workshop on In Situ Data Management (ISDM) on January 28-29, 2019 at the Bethesda North Marriott in Rockville, MD. This document provides background information on ISDM, information about the purpose of workshop and expected outcomes, as well as some logistics information for participants.

Purpose of the Workshop

Scientific computing will increasingly incorporate a number of different tasks that need to be managed along with the main simulation tasks. For example, this year's SC18 agenda featured in situ infrastructures, in situ analytics, big data analytics, workflows, data intensive science, machine learning, deep learning, and graph analytics—all nontraditional applications unheard of in an HPC conference just a few years ago. Perhaps most surprising, more than half of the 2018 Gordon Bell finalists featured some form of artificial intelligence, deep learning, graph analysis, or experimental data analysis in conjunction with or instead of a single computational model that solves a system of differential equations.

We define ISDM as the practices, capabilities, and procedures to control the organization of data and enable the coordination and communication among heterogeneous tasks, executing simultaneously in an HPC system, cooperating toward a common objective. This workshop considers In Situ Data Management beyond the traditional roles of accelerating simulation I/O and visualizing simulation results, to more broadly support future scientific computing needs. We seek to identify priority research directions for ISDM to support current and future HPC scientific workloads, which include, for example, the convergence of simulation, data analysis, and artificial intelligence, requiring machine learning, data manipulation, creation of data products, assimilation of experimental and observational data, analysis across ensemble members, and, eventually the incorporation of tasks on non-von Neumann architecture.

The Potential for Greater Scientific Impact from In Situ Data Management

Simulations on HPC systems can generate data up to five orders of magnitude greater than the maximum data volume that can be exported to the storage system. Current approaches to managing this bottleneck focus on executing data analysis and visualization tasks in situ—within the HPC system itself—to produce data products that can be orders of magnitude smaller than the full state data. The visualization and I/O communities have developed a range of in situ data processing and analysis technologies as a way of achieving data analysis capabilities despite I/O bottlenecks on HPC systems.

Managing data in situ, that is, processing data while they are being generated, can lead to better use of storage resources and better science: it eliminates some of the negative impacts of the I/O bottleneck; saves storage space; allows the data analysis and processing tasks to access the full data from the simulation as opposed to just the output data; reduces data movement; and reduces time to solution, to name a few. In fact, in a growing number of cases, in situ data techniques are the only way to process and analyze data. The in situ paradigm, however, also complicates some operations. For example, human interaction, exploratory investigation, and temporal analysis may be easier to conduct post hoc. Hence, there is a rich design space for carrying out computation in situ: determining which data products are needed for post hoc analysis and the graph of in situ tasks needed to create these; scheduling and executing in situ tasks; and managing the data and communication flow among these tasks.

A motivation for this workshop is that ISDM capabilities could be expanded and leveraged for a broader range of current and future HPC applications. In addition to helping meet the challenges of extreme-scale simulation data, ISDM technologies can facilitate applications that merge simulation and data analysis, simulation and machine learning, or the processing and analysis of experimental data.

Workshop Deliverables

The primary outcome of the workshop is a short list of (typically 3-5) high-level *priority research directions* (PRDs). A PRD is an articulation of community-level research goals synthesized from the ideas generated during the workshop. The workshop agenda[1] is divided into several breakout sessions, the topics for which were chosen to cover the relevant technical areas and elicit productive suggestions from the participants. However, *ideas and suggestions should not be limited by the breakout session topics of focus.* PRDs often transcend and cross-cut multiple topics; and ideas generated by different breakout sessions may point to the same PRD.

Each breakout session will be required to develop a number of candidate PRDs, to be presented in the read-out sessions on the second day (Figure 1 and Figure 2). The organizing committee will produce draft PRDs for the workshop based on the breakout sessions' candidate PRDs. These will be presented to and discussed by all workshop participants during the final plenary session on the second day, during which changes can be proposed and considered. A

[1] See Appendix for the workshop agenda.

detailed description of the final PRDs, along with summaries of the individual topic discussions, will appear in a written report to follow after the workshop.

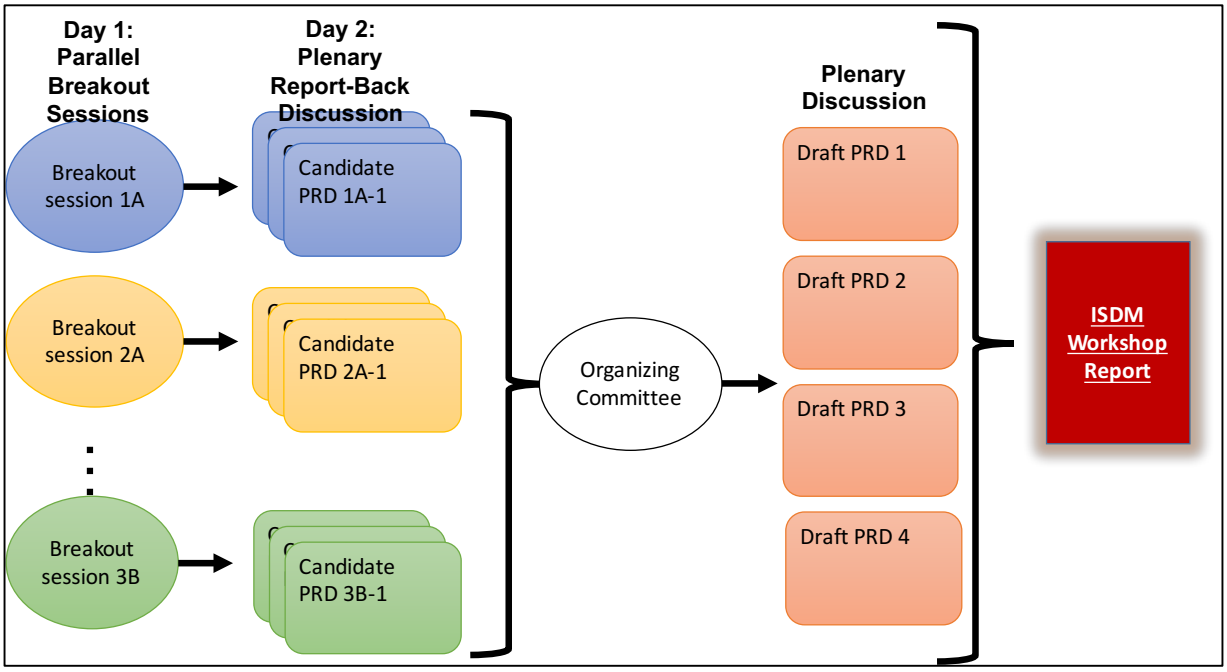

Figure 1: Candidate PRDs identified during the breakout sessions will be read back to the entire workshop during the second day and prioritized.

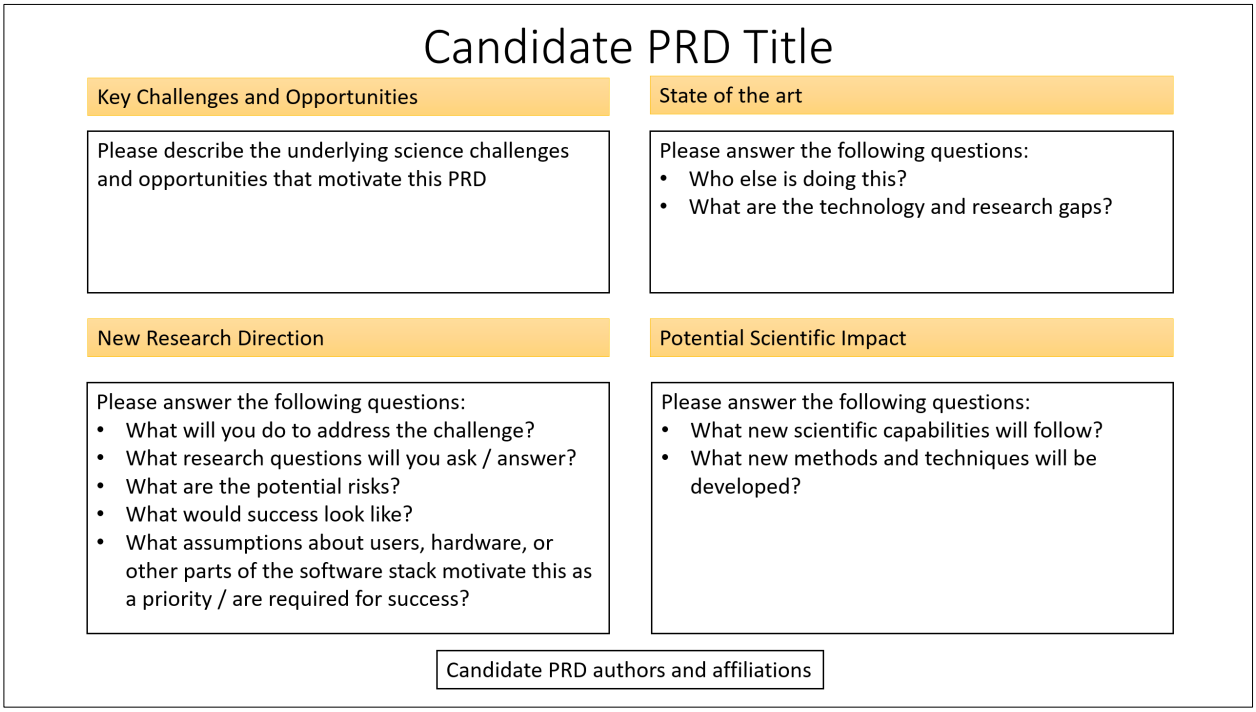

Figure 2: PRD quad chart template.

2. Summary of Present State

In Situ Technology Past and Present

A survey of past and present in situ methods and tools[2] demonstrates how reusable in situ software evolved separately from the storage and visualization communities. Storage solutions originally were used for staging a simulation's state for checkpointing, restarting, or saving outputs for later post hoc analysis. Even though such tools have expanded their applications beyond I/O staging, their I/O style of interface and data model remain. Meanwhile, the scientific visualization community developed in situ equivalents of their post hoc tools. Coming at the in situ problem from a visualization direction, these tools feature the VTK data model and scripts for connecting and executing pipelines of VTK filters.

Today, new tools are being developed for more generic data producer / consumer tasks with the potential to manage a general graph of tasks communicating custom data types. There lacks, however, a common vision for core capabilities to be delivered to users; as well as sufficient attention to making these tools interoperable. To more broadly support scientific computing needs, this workshop will provide a forum to address generic in situ data management capabilities, for example for machine learning, automated spawning of ensemble runs, automated triggering and production of data products, and tasks run on non-von Neumann architectures. There are also opportunities to discuss provenance and uncertainty as data are managed across tasks, as well as facilitating workflows across multiple data and computing resources through interfaces between distributed and in situ workflows systems.

Science Drivers

This workshop anticipates a future of diverse HPC workloads that will increasingly include the application types listed below for which in situ data management provides enabling capabilities. The following are illustrative examples, not intended to be all-inclusive, of current or future uses of ISDM.

- Smart simulations featuring online feedback and potential computational steering
- Ensemble analysis of stochastic or rare events, uncertainty studies, or model calibration
- High-fidelity, highly scalable data analysis and visualization
- Workflows featuring the convergence of "Big Data" and HPC software and tools
- Use of machine learning and deep learning alongside simulations or experiments
- Real-time experimental and observational data analysis and assimilation

[2] Bauer et al. In Situ Methods, Infrastructures, and Applications on High Performance Computing Platforms. Proceedings of EuroVis 2016 Conference, 2016.

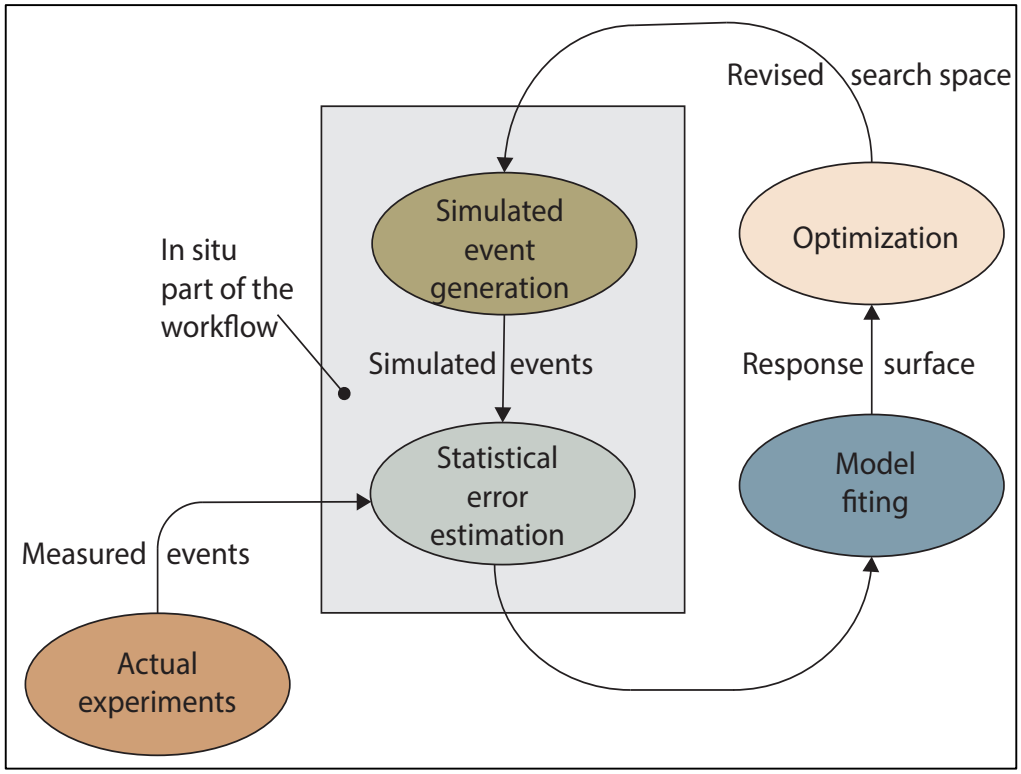

Figure 3: HEP workflow of in situ neutrino event generation and parameter optimization.

Two examples of applications embodying some of the above elements are illustrated in Figure 3 and Figure 4. Figure 3 is a future vision of smart simulations in an HEP-ASCR partnership.[3] The goal of this activity is to automatically simulate millions of concurrent Monte Carlo proton-proton collisions, search the response surface for minimum statistical difference with experimentally observed collisions, and advance toward new regions of the parameter space to investigate. Figure 4 is an example of ensemble analysis. It shows a BES-ASCR collaboration to simulate nucleation as a material cools and crystallizes.[4] I/O bottlenecks are avoided by detecting crystal structures in situ and only storing features of interest. Instead of one large simulation, many smaller instances are launched dynamically until a rare event is detected; a pattern that has widespread applicability to other domains such as protein folding, self-assembled structures, and genetic algorithms.

[3] Norman et al. Implementation of Feldman-Cousins Correction and Oscillation Calculations in the HPC Environment for the NOvA and DUNE Experiments. Proceedings of CHEP 2018 Conference, 2018.
[4] Yildiz et al. Heterogeneous Hierarchical Workflow Composition. Submitted to Computers in Science and Engineering (CiSE) Journal Special Issue on Scientific Workflows, 2019.

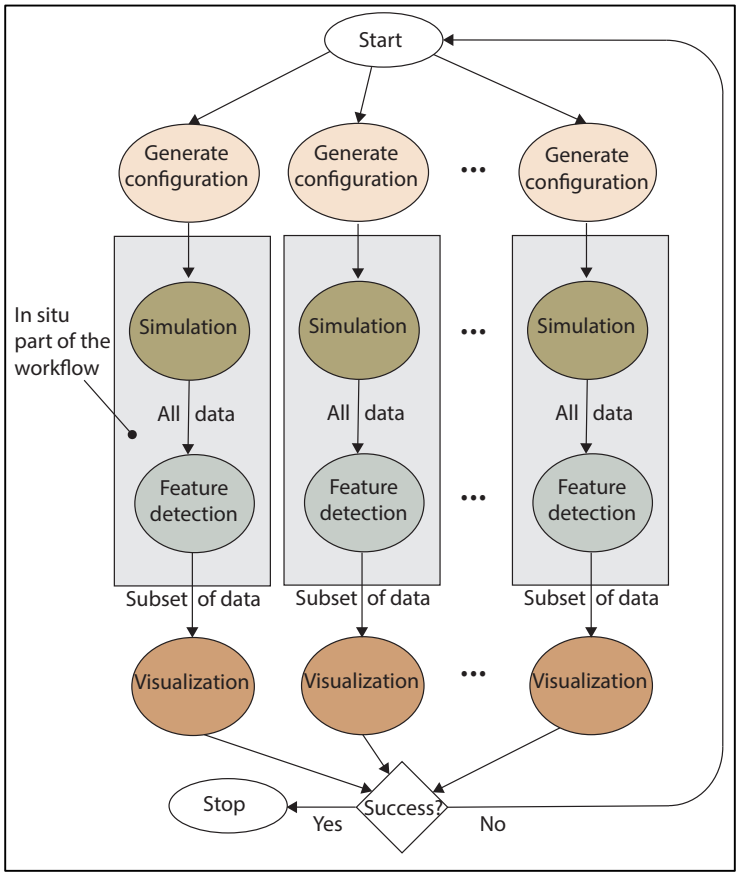

Figure 4: BES workflow of dynamic ensemble of simulations and in situ detection of stochastic events.

Computing Ecosystem

The end of Moore's law and Dennard scaling has led to increased concurrency and heterogeneity in computing units and reliance on both general and dedicated-purpose accelerators. Disparity in data movement latency, bandwidth, and energy consumption compared with the rate of floating point operations has led to deeper memory and storage hierarchies. To put the imbalance between computing and data management in perspective, the rate of data that can be computed on the Summit supercomputer (assuming 1 byte generated per clock cycle) is five orders of magnitude greater than the bandwidth of its parallel file system. The I/O bottleneck is one driver of the need for in situ analysis.

Current approaches to manage this bottleneck focus on executing data analysis, visualization, and the production of data products in situ. The resulting data products are often orders of magnitude smaller than the full state data, thereby eliminating some of the negative impacts of the I/O bottleneck and saving storage space. In situ analyses can also lead to better science. While the infrequency of data outputs limits the fidelity of post hoc analysis, in situ analysis can have much higher fidelity because analysis tasks have access to simulation data directly, and are not throttled by the I/O. The in situ paradigm, however, also complicates some operations. For example, human interaction, exploratory investigation, and temporal analysis are easier to

conduct post hoc. In situ methods also add complexity to the workflow because of the larger number of interconnected, concurrent tasks that need to be managed.

3. Outline of the Workshop

The workshop agenda appears in the appendix. Day 1 of the workshop begins with a plenary session and proceeds to breakout sessions on the following ISDM topics:

- science applications,
- computational platforms,
- data and communication models,
- programming and execution models,
- provenance and reproducibility,
- analysis algorithms, and
- software architecture.

Each of these topics is explained below.

The plenary session begins with talks from application domain scientists and is intended to help workshop participants think about ISDM in new ways. Following are 6 breakout groups organized into 3 sessions of 2 parallel breakouts each. Each breakout group features a topic area described below, with the goal of each breakout being to submit several candidate PRDs for consideration on the second day of the workshop.

Day 2 begins with short informative talks about other related workshops, with the remainder of the day being dedicated to distilling ideas from day 1 into draft PRDs for the workshop report. We envision that each breakout will report several equally-viable candidate PRDs. The final session of the workshop is an all-group discussion where we will prioritize, synthesize, and cull the candidate PRDs down to a final set of 3 to 5 draft PRDs to be included in the workshop report.

The following one-page descriptions of each of the breakout sessions are intended to provide some context for each topic and to start the conversation with some leading questions, but they are not intended to limit discussion to the points listed. We have also identified some questions that are common to all topics, and we would like participants to consider and address the following in each session:

- What assumptions and dependencies does this topic have on other topics in the ISDM workshop?
- What interactions and linkages does this topic have with research in other ASCR workshops and programs?

Science Applications

Goals of ISDM include enabling useful and insightful in situ analysis at the desired level of fidelity in order to provide feedback to the user, or to automatically steer the simulation or analysis. “Users” here are broadly defined as simulation scientists, experimentalists, computer scientists, and application developers. Users often want ISDM to provide efficient utilization of in situ data for analysis with little impact on running simulations or disruption of streaming input data. ISDM can decrease the copying and conversion of in situ algorithm input and output data, provide appropriate data structures that capture necessary information, improve computational performance of in situ algorithms within the application, and potentially provide provenance and resilience. The scientific applications topic area explores commonalities among applications and examines categories of use cases that drive many of the other topic areas of the workshop, including the state of the practice that will identify gaps in ISDM.

Overall driving question: What ISDM capabilities are needed to best meet science application requirements? Such requirements can be found in the *Exascale Requirements Review*.[5]

Subtopic: Commonalities of current and future applications using in situ analysis in similar ways

- Can we find a useful categorization of scientific applications using in situ analysis?
- Are there common data models and/or programming models?
- What are the typical in situ analysis algorithms and in situ frameworks used for each application category?

Subtopic: In situ frameworks and usability

- What are the missing elements in existing frameworks that prevent wider adoption? How will frameworks need to evolve to meet the needs of science applications in the future? New architectures?
- What skills are needed by science teams to develop frameworks/technology? What is needed in workforce development to address the needs of future applications? Where is the education gap?
- What is required of an application to use in situ analysis algorithms?
- What are the tradeoffs between domain-specific and generic in situ frameworks?
- Do performance and scalability of in situ frameworks affect adoption? If so, in what way?

Subtopic: ISDM for the science application workflow

- How can ISDM support a science application workflow that includes both in situ and distributed-area or inter-facility components?
- How can workflow management systems (WMSs) couple to ISDM tools for in situ analysis?

[5] http://exascaleage.org/

Data Models: Connection and Communication

This workshop views data models as abstractions and implementations describing how a set of values in memory should be interpreted as a relevant scientific object, as well as the middleware and/or communication tools needed to access data in situ. As such, tools to register, manipulate, communicate, publish, and query data models at multiple levels are key to moving beyond having each in situ component being a bespoke, one-off implementation. Some communities have adopted a single unified data standard that allows them to implicitly address data model matching issues. Similarly, some programming models (see Programming and Execution Models section) enforce particular data approaches like zero-copy pointer access that enable simpler runtime assumptions but may sacrifice usage flexibility.

Both of these approaches can limit programmer productivity and software reusability, however. More explicit and robust tools, methods, and frameworks are required to improve ISDM data descriptions and communications without providing undue burden on the programmer or end user. Hence, data models for ISDM cover several overlapping issues: structural definition (Is it an integer? A 64-bit floating point?), semantic definition (Does this linked list represent a graph?), and access definition (Is this serialized as a message or do pointers access scattered memory locations?).

Overall driving question: Are there data model commonalities or motifs for description and access that we can identify that will promote programmer productivity and software reusability?

Subtopic: Data interchange

- What framework or tools services could automate the conversion between differing producer and consumer data models?
- How should we best address the mismatch between producer and consumer data model definitions?

Subtopic: Performance and usability of data models

- What is the interplay between data model and data communication in light of evolving heterogeneity in systems and performance portability?
- What developments are needed to make data format management, zero-copy data layout descriptions, and/or automated structure definition more universally available?
- How do data model access and descriptor models change for accelerators?

Subtopic: Metadata

- What is the interplay between computational, provenance, performance portability, and archival data models? Are there differences in time scale where they are relevant?
- Do we need metadata schema for analysis, visualization, deep learning, and other in situ components, or are there schema-less approaches that offer advantages?

Computational Platforms and Environments

While previous in situ efforts reacted, in a defensive manner, to changes in computational platforms and environments (the I/O bottleneck, for example), a more strategic approach is needed to ensure that platforms and environments meet the analysis and visualization needs of computational scientists, and that in situ software is flexible enough to exploit emerging technologies. In this workshop, we elicit ideas that exploit recent and anticipated changes to high-performance computational platforms and environments as opportunities for the in situ analysis research community. For example, nonvolatile memory is expanding per-node storage capacity, affording potential opportunities for creative data management and new analysis techniques; and new computational platforms are designed to support machine learning. In addition, we expect somewhat pervasive processing capabilities through complex heterogeneous node architectures, such as systems with GPUs, FPGAs, processing-in-memory, processing-in-network, and neuromorphic hardware. Such advances create exciting opportunities for in situ analysis research.

Overall driving question: How do we develop ISDM technologies to adequately exploit emerging computational platform and environment capabilities?

Subtopic: Memory and storage architectures

- How should ISDM technologies exploit/influence SSIO innovations in storage and data management, for example multi-level memory, NVRAM, and in-system data services? What interfaces with storage system software are advantageous?

Subtopic: Heterogeneous node architectures and pervasive computing

- What opportunities for ISDM arise from the computing characteristics of GPUs, FPGAs, neuromorphic hardware, and processing-in-memory? What level of portability can be ensured for ISDM capabilities? How?

Subtopic: Operating system and runtime requirements

- What are the operating system and runtime (OS/R) requirements for in situ analysis?
- How should ISDM technologies share resources (e.g., memory, storage, and accelerators) among tasks and with other parts of the software stack (e.g., application, file system, runtime system)?
- What advances in OS/R technology will be needed to fully realize a convergence of HPC and “Big Data” analysis, such as machine learning, experimental data streaming, and others? How should ISDM capabilities interface with or influence these new technologies?

Subtopic: Role of ISDM in co-design

- Does ISDM put unique stresses on hardware or contribute unique needs not already represented by other use cases? If yes, how should ISDM be considered by HPC system designers?

Analysis Algorithms

Data analysis algorithms that operate in situ may have unique characteristics and requirements compared with those designed for post hoc execution. Algorithms designed for in situ execution can potentially be scaled to the full spatiotemporal resolution of data being produced by the application, at the rate being produced, and on platforms of extreme concurrency and significant heterogeneity. To be effective in this regard, advances in scalable and platform-portable methods is a high priority.

Overall driving question: How do we architect analysis algorithms, including ML, to be scalable, platform-portable, and to support the increased diversity and complexity of in situ science use cases on future generations of computational platforms?

Subtopic: Performance portability

- How can performance and scalability across applications (computational and experimental), workflows (in situ and distributed), and heterogeneous architectures (current and future) be achieved?

Subtopic: Data-driven algorithms and emerging in situ use cases

- How can existing algorithms in “Big Data” tools and frameworks be used, and which algorithms must be redesigned? How should scalable, parallel, “explainable” machine learning algorithms that obey physical models and/or constraints be developed?
- In the absence of human interactivity, how can parameters “intelligently” be set in computationally-steered workflows?
- What algorithmic challenges arise in complex in situ workflows (e.g., real-time model calibration, integration of experimental, observational, and simulation data, and high-fidelity uncertainty quantification)?
- What new opportunities exist for analysis methods when working with full spatiotemporal resolution data? What new science is possible?

Subtopic: Resource-constrained and approximate methods

- How can low-complexity approximate solution techniques, including sampling approaches, surrogates, and/or reduced-order models be used?
- Can analysis algorithms be redesigned to minimize data movement, energy, or conserve other resources (e.g., communication-avoiding algorithms or stochastic communication)?

Subtopic: Relationship between ISDM framework and algorithmic design

- What algorithmic primitives should an ISDM framework provide? What can/should be borrowed from “Big Data” frameworks (e.g., Spark’s reduce by key)?
- What is the interplay between ISDM frameworks, algorithms, and data models? What co-design challenges exist?
- How can ISDM be an enabler and not a barrier to developing performance-portable, sustainable, and interoperable algorithms? How can ISDM enable sharing and simplify development of new methods?

Provenance and Reproducibility

ISDM creates a degree of separation between scientists and data, making provenance essential in order to trust the results of in situ workflows. For example, with the increasing adoption of machine learning in ISDM, capturing hyperparameters, noise levels, decision points, and training data can facilitate replicating results and increase confidence in predictions. Provenance—a record of data products and their transformations—serves multiple purposes: trust, code debugging and optimization, data quality and audit, and scientific reproducibility. The potential volume and verbosity of provenance information, the cost of capture, and the complexity of supporting metadata make a principled, targeted, and systematic approach to in situ provenance collection essential.

Overall driving question: For in situ applications, what is the minimum set of provenance information that needs to be extracted so that captured information will be useful for various purposes later?

Subtopic: Uses of in situ provenance information

- How will provenance information collected in situ be used post hoc?
- How can provenance enable or improve search, trust or quality assurance, performance analysis, and/or reproducibility?

Subtopic: Targeted in situ collection of provenance

- What are the application- and architecture-dependent goals for which in situ provenance needs to be collected?
- To what extent should ISDM software and frameworks support the selection, capture, interpretation, and usage of provenance information and metadata?
- To what extent should they support data reduction of provenance information?

Subtopic: Provenance for performance analysis

- What impact will provenance collection have on in situ execution, given that resources are already shared between various tasks?
- What are design criteria for minimally invasive provenance extraction and maximum usability in ISDM?
- What are the tradeoffs between enabling reproducibility and potent impact on application performance and resource usage?

Subtopic: Provenance for result validation, replication, and reproducibility

- What does reproducibility mean for in situ workflows where data are transformed for analysis and only derived data products are available post hoc?
- What is the minimal provenance required for a data product created in situ to be reproducible and reusable? How does that differ from products created post hoc?
- If machine learning is used for in situ analysis, what provenance should be extracted to ensure trust and reproducibility of results? How will it differ from provenance information for machine learning applications post hoc?

Programming and Execution Models

Programming and execution models (PEMs) are critical to the discussion of ISDM, because the manner in which in situ data are accessed and managed across HPC resources largely depends on the programming model used. The PEM also controls how in situ computations are run (parallelism and ordering of computations) and where they are run (what node and what processing unit). Often PEMs include data models or have constraints on how data are managed (see the Data Models: Connections and Communication section).

Examples of PEMs that could be included in this topic area include but are not limited to 1) Bulk synchronous and asynchronous models, 2) Task-based models, 3) “Big Data” (databases, message queueing systems, map-reduce, etc.) 4) Mixed models (eg. MPI + X) or converged models (“Big Data” + HPC)

Overall driving question: How can PEMs support ISDM for effective and efficient in situ data analysis?

Subtopic: Suitability of PEMs for in situ data management

- Are there aspects of PEMs that best support certain types of in situ data analysis or in situ data types?
- How can in situ data analysis be supported within systems for ISDM workflows that may use multiple PEMs and/or converged “Big Data” + HPC PEMs? Are there any special considerations for ISDM in this case?
- Is there a long-term, performance-portable PEM solution that does not require a code rewrite every 5 years?

Subtopic: Support for dynamic computations or data generation that produce data at different rates and on different resources

- How can PEMs support dynamic in situ data analysis for irregular or unpredictable input data generation?
- How can PEMs support performance-portable in situ data analysis? How can they support visibility into the performance tradeoffs of in situ data analysis?
- How can PEMs keep up with input data generation, especially for applications that need to be real-time or pseudo real-time?

Subtopic: Usability of PEMs for in situ analysis

- Are there any domain-specific high-level PEM interfaces that might better enable in situ analysis frameworks or in situ analysis, or be more usable for varying types of users and levels of user expertise?
- How can PEMs support provenance capture during in situ analysis with the goal of understanding the lineage of the results?
- How can PEMs support resilience when in situ computations may fail?

Software Architecture for Usability and Sustainability

The ISDM "software environment" is diverse and multi-faceted; this topic area examines the design of high-level software architecture that promotes the usability, utility, and longevity of ISDM methods and infrastructure. In the ISDM context, the software environment may be thought of enabling the execution of a dynamic graph of multiple producers and consumers. Producers are sources of data, and may be simulations, experiments, or combinations of both. Consumers ingest and process data, and may in turn become data producers for other downstream consumers. The graph, which defines processing order, may be dynamic and change in response to data- or processing-dependent factors. The set of nodes in the graph, the tasks (see the Analysis Algorithms section), may be diverse, and may be scheduled for execution on different types of hardware (see the Computational Platforms and Environments section).

Overall driving question: What design issues will promote longevity of ISDM software as well as promote adoption and use by the science community?

Subtopic: ISDM software and ecosystem architecture

- What would an "ideal" ISDM ecosystem look like? From a researcher/developer point of view, from a user's point of view, and from an ASCR research portfolio view?
- What design and architecture choices promote reuse and inclusion of software tools from many sources, both ASCR-funded research efforts as well as 3rd party tools, in a composable fashion, such that an ISDM software stack/ecosystem maximizes use of technology from diverse sources?

Subtopic: Usability and adoption

- What are the pros and cons of using ISDM software infrastructure/tools as opposed to deploying bespoke technologies and methods directly into an application?
- What types of software users/developers will interact with, extend, develop, and deploy ISDM technologies?
- From the user's point of view, what are impediments to using and usability of ISDM software, and how should these issues be addressed? (Users may be application, library, or analysis developers.)
- What does reusability mean in this context, and what are the barriers to reusability?

Subtopic: Sustainability and growth

- What are impediments to the sustainability of ISDM software methods and tools, and what research is needed to address these issues?
- What are advantages of using 3rd party, emerging capabilities, such as machine learning tools, and alternate design/execution patterns, as part of the ISDM approach?
- What does ISDM software research, development, and deployment (R&D&D) need from other areas, like programming models, OS/R, operational policy at HPC centers, distributed workflows, data streaming, etc.?

4. Appendix

Agenda

Monday, January 28, 2019

Time	Activity	Activity
7:30 AM – 8:30 AM	Registration and breakfast available	
8:30 AM - 9:15 AM	Opening remarks	
9:15 AM - 10:45 AM	Plenary session: Science applications	
10:45 AM - 11:00 AM	Break	
11:00 AM - 12:30 PM	Breakout session 1A: Data models: connection and communication	Breakout session 1B: Computational platforms and environments
12:30 PM - 1:30 PM	Lunch	
1:30 PM - 3:00 PM	Breakout session 2A: Analysis algorithms	Breakout session 2B: Provenance & reproducibility
3:00 PM - 3:30 PM	Break	
3:30 PM - 5:00 PM	Breakout session 3A: Programming & execution models	Breakout session 3B: Software architecture for usability and sustainability

Tuesday, January 29, 2019

Time	Activity
7:30 AM – 8:30 AM	Registration and breakfast available
8:30 AM - 9:15 AM	Summaries of related workshop activities
9:15 AM - 10:00 AM	Report back from breakout sessions 1A and 1B
10:00 AM - 10:45 AM	Report back from breakout sessions 2A and 2B
10:45 AM - 11:15 AM	Break
11:15 AM - 12:00 PM	Report back from breakout sessions 3A and 3B
12:00 PM - 1:00 PM	Lunch
1:00 PM - 2:30 PM	Prioritizing research directions
2:30 PM	Workshop adjourns

Table 1: Workshop agenda

Organizing Committee

Name	Affiliation	Role	Email
Tom Peterka	Argonne National Laboratory (ANL)	Chair	tpeterka@mcs.anl.gov
Debbie Bard	National Energy Research Scientific Computing Center (NERSC)	Organizer	djbard@lbl.gov
Janine Bennett	Sandia National Laboratories (SNL)	Organizer	jcbenne@sandia.gov
Wes Bethel	Lawrence Berkeley National Laboratory (LBNL)	Organizer	ewbethel@lbl.gov
Ron Oldfield	Sandia National Laboratories (SNL)	Organizer	raoldfi@sandia.gov
Line Pouchard	Brookhaven National Laboratory (BNL)	Organizer	pouchard@bnl.gov
Christine Sweeney	Los Alamos National Laboratory (LANL)	Organizer	cahrens@lanl.gov
Matthew Wolf	Oak Ridge National Laboratory (ORNL)	Organizer	wolfmd@ornl.gov
Laura Biven	U.S. Department of Energy Advanced Scientific Computing Research (DOE-ASCR)	Program Manager	laura.biven@science.doe.gov

Table 2: Workshop organizing committee

Related Workshops and Activities

- Findings of the ASCR Basic Research Needs Workshop on Scientific Machine Learning (January 2018)
    - DOI:10.2172/1484362
- Findings of the ASCR Basic Research Needs Workshop on Extreme Heterogeneity (January 2018)
    - DOI: 10.2172/1473756
- Dagstuhl In Situ Visualization for Computational Science Workshop (July 2018)
    - https://www.dagstuhl.de/en/program/calendar/semhp/?semnr=18271
- ASCR Storage Systems and Input/Output Workshop (September 2018)
    - https://www.orau.gov/ssioworkshop2018/default.htm
- Gap Analysis: Materials Discovery through Data Science at Advanced User Light Sources Workshop (October 2018)
    - http://www.cvent.com/events/gap-analysis/event-summary-3ad86ec662904f829af44a6f24f14dc7.aspx
- NITRD Convergence of High Performance Computing, Big Data, and Machine Learning Workshop (October 2018)
    - https://www.nitrd.gov/nitrdgroups/index.php?title=HPC-BD-Convergence
- ASCR Exascale Requirements Review workshop reports
    - http://exascaleage.org/